(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 117 207 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
18.07.2001 Bulletin 2001/29

(51) Int Cl.[7]: **H04L 9/32**

(21) Application number: **01300224.1**

(22) Date of filing: **11.01.2001**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE TR
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **14.01.2000 US 483356**

(71) Applicant: **Hewlett-Packard Company**
Palo Alto, California 94304-1112 (US)

(72) Inventor: **Corella, Francisco**
Hayward, California 94541 (US)

(74) Representative: **Powell, Stephen David et al**
WILLIAMS, POWELL & ASSOCIATES,
4 St Paul's Churchyard
London EC4M 8AY (GB)

(54) Public key infrastructure

(57) A PKI (30) includes an off-line registration authority (38) that issues a first unsigned certificate (60) to a subject (34) that binds a public key (62) of the subject to long-term identification information (63) related to the subject and maintains a certificate database (40) of unsigned certificates in which it stores the first unsigned certificate. An on-line credentials server (42) issues a short-term disposable certificate (70) to the subject that binds the public key of the subject from the first unsigned certificate to the long-term identification information related to the subject from the first unsigned certificate. The credentials server maintains a table (44) that contains entries corresponding to valid unsigned certificates stored in the certificate database. The subject presents the short-term disposable certificate to a verifier (36) for authentication and demonstrates that the subject has knowledge of a private key corresponding to the public key (46) in the short-term disposable certificate.

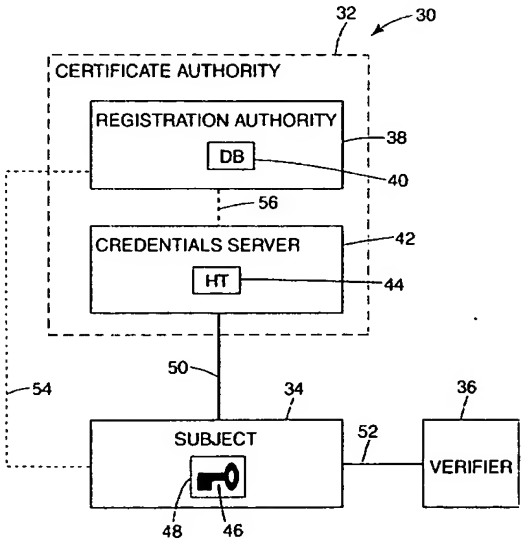

Fig. 1

EP 1 117 207 A2

Description

[0001] The present invention relates to public key cryptosystems, and more particularly, to a lightweight public key infrastructure employing disposable short-term certificates for authentication and/or authorization.

[0002] Public key cryptosystems are globally deployed on the World Wide Web, as well as on a growing number of enterprise networks, for establishment of secure communication channels. Every user in a public key cryptosystem has a pair of keys including a public key and a private key. The public key is disclosed to other users while the private key is kept secret. A public key cryptosystem typically has a primary designed use, such as for encryption, digital signature, or key agreement. Public key cryptosystems are also used for user authentication. For example, a user can authenticate itself to other users by demonstrating knowledge of its private key, which other users can verify using the corresponding public key.

[0003] In an application of a public key cryptosystem for authenticating a user, the public key must be securely associated with the identity of the user that owns the public key by authenticating the public key itself. Public key certificates are typically employed to authenticate the public key. A public key certificate is a digital document, signed by a certificate authority, that binds a public key with one or more attributes that uniquely identify the owner of the public key. The public key certificate can be verified using the certificate authority's public key, which is assumed to be well known or is recursively certified by a higher authority. For example, in a corporation, a public key certificate can bind a public key to an employee number.

[0004] A public key infrastructure (PKI) refers to the collection of entities, data structures, and procedures used to authenticate public keys. A traditional PKI comprises a certificate authority, public key certificates, and procedures for managing and using the public key certificates.

[0005] One type of a user of a PKI owns the public key contained in a public "key certificate and uses the certificate to demonstrate the user's identity. This type of user is referred to as the subject of the certificate or more generally as the subject. Another type of user relies on a public key certificate presented by another user to verify that the other user is the subject of the certificate and that the attributes contained in the certificate apply to the other user. This type of user that relies on the certificate is referred to as a verifier or relying party.

[0006] The association between a public key and an identity can become invalid because the attributes that define the identity no longer apply to the owner of the public key, or because the private key that corresponds to the public key has been compromised. A PKI typically employs two complementary techniques for dissociating a public key from an identity. In the first technique, each public key certificate has a validity period defined by an expiration date, which is a substantial period from the issue date, such as one year from the issue date. In the second technique, the certificate authority revokes a public key certificate if the public key certificate's binding becomes invalid before the expiration date. One way of revoking a public key certificate is by including a serial number of the public key certificate in a certificate revocation list (CRL), which is signed and issued by the certificate authority at known periodic intervals, such as every few hours or once a day. An entity that relies on a certificate is responsible for obtaining the latest version of the CRL and verifying that the serial number of the public key certificate is not on the list.

[0007] CRLs typically become quite long very quickly. When the CRLs become long, performance is severely impacted. First, CRL retrieval consumes large amounts of network bandwidth. Second, each application has to retrieve the CRL periodically, parse the CRL, and allocate storage for the CRL. Then, the application needs to carry out a linear search of the CRL for the public key certificate serial number when the application verifies each public key certificate. As a result, conventional PKIs do not scale beyond a few thousand users.

[0008] One solution proposed to alleviate the linear search problem is to partition CRLs. The serial number of the public key certificate determines where the CRL partition is located when the public key certificate is revoked. With partitioned CRLs, the application still has to retrieve and store the entire CRL or else the application needs to fetch a CRL partition in order to verify a certificate. Since certificate verification is a likely critical path, fetching a CRL partition impacts the time it takes to run the application.

[0009] An on-line certificate status protocol (OCSP) operates by permitting the verifier of the public key certificate to ask the certificate authority if the certificate is currently valid. The certificate authority responds with a signed statement. The OCSP allows CRLs to be avoided, but requires the verifier to query the certificate authority as part of the transaction that employs the public key certificates. The verifier querying the certificate authority increases the time it takes to perform the transaction. The OCSP scheme is highly vulnerable to a denial-of-service attack, where the attacker floods the certificate authority with queries. Responding to each query is computationally expensive, because each response requires a digital signature.

[0010] In a certificate status proofs scheme, the certificate authority maintains a data structure describing the set of valid and invalid certificates in the directory. For every public key certificate that has not yet expired, a short cryptographic proof can be extracted from the data structure of the certificate's current status (i.e., valid or invalid). A CRL can essentially be viewed as a cryptographic proof of invalidity for the public key certificates in the CRL, and proof of validity for those not in the CRL. The CRL, however, is not a short proof. The short cryptographic proof can be obtained by the verifier from the

directory, or it can be obtained by the subject and presented to the verifier together with the public key certificate.

[0011] The Simple Public Key Infrastructure (SPKI) working group of the Internet Society and the Internet Engineering Task Force has proposed the possibility of employing short-lived certificates as a method of achieving fine-grain control over the validity interval of a certificate. See C.M. Ellison, B. Frantz, B. Lampson, R. Rivest, B.M. Thomas and T. Ylonen, *SPKI Certificate Theory*, Request for Comments 2560 of the Internet Engineering Task Force, September 1999. The *SPKI certificate theory* reference states that there are cases in which a short-lived certificate requires fewer signatures and less network traffic than various on-line test options. The use of a short-lived certificate always requires fewer signature verifications than the use of a certificate plus on-line test result.

[0012] Nevertheless, no practical method of issuing short-lived certificates has been proposed. Traditional certificates are issued off-line, as part of a process that includes subject registration, at the rate of one per year per user. By contrast, short-lived certificates would have to be issued on-line at the rate of at least one per day per user, and perhaps as often as one every few minutes for every user.

[0013] The term on-line and the term off-line have particular definitions in the context of a PKI. The term on-line herein refers to the day-to-day usage of public key certificates and key pairs for authentication. The term off-line herein refers to the more infrequent establishment or dissolution of public key bindings, which may result in the issuing or revocation of public key certificates. For example, the traditional certificate authority is off-line, issues CRLs off-line, and places the CRLs in a directory for on-line retrieval. The scheme involving certificate status proofs also makes use of off-line certificate authorities. The OCSP is the only scheme described above that employs an on-line certificate authority.

[0014] For reasons stated above and for other reasons presented in greater detail in the Description of the Preferred Embodiment section of the present specification, there is a need for an improved lightweight PKI that overcomes the above-described revocation problems and efficiently scales well beyond a few thousand users, and which can be employed for authentication and/or authorization.

[0015] The present invention provides a public key infrastructure (PKI) including a subject, an on-line registration authority, an off-line credentials server, and a verifier. The registration authority issues a first unsigned certificate off-line to the subject that binds a public key of the subject to long-term identification information related to the subject. The registration authority maintains a certificate database of unsigned certificates in which it stores the first unsigned certificate. The credentials server issues a short-term disposable certificate on-line to the subject. The short-term disposable certificate binds the public key of the subject from the first unsigned certificate to the long-term identification information related to the subject from the first unsigned certificate. The credentials server maintains a table that contains entries corresponding to valid unsigned certificates stored in the certificate database. The subject presents the short-term disposable certificate to the verifier for authentication and demonstrates that the subject has knowledge of a private key corresponding to the public key in the short-term disposable certificate.

[0016] Figure 1 is a block diagram of a light-weight public key infrastructure (PKI) according to the present invention employing short-term disposable certificates.

[0017] Figure 2 is a diagram of an unsigned certificate issued from a registration authority of the PKI of Figure 1.

[0018] Figure 3 is a flow diagram of an off-line protocol for issuing an unsigned certificate from a registration authority of the PKI of Figure 1.

[0019] Figure 4 is a diagram of a non-structured short-term disposable certificate as issued by a credentials server of the PKI of Figure 1.

[0020] Figure 5 is a flow diagram illustrating an on-line protocol for issuing a short-term disposable certificate from a credentials server of the PKI of Figure 1.

[0021] Figure 6 is a flow diagram illustrating an on-line protocol-for a subject to demonstrate its identity to a verifier of the PKI of Figure 1.

[0022] Figure 7 is a flow diagram illustrating a protocol for a registration authority to revoke an unsigned certificate for the PKI of Figure 1.

[0023] Figure 8 is a block diagram of a light-weight PKI employing short-term disposable certificates for authorization according to the present invention.

[0024] Figure 9 is a diagram of a structured short-term disposable certificate as issued by a credentials server of the PKI of Figure 8.

[0025] Figure 10 is a flow diagram illustrating an on-line protocol for issuing a structured short-term disposable certificate from a credentials server of the PKI of Figure 8.

[0026] Figure 11 is a flow diagram illustrating an on-line protocol used by a subject to demonstrate its identity to a verifier of the PKI of Figure 8.

[0027] Figure 12 is a block diagram of a distributed certificate authority having distributed credentials servers according to the present invention.

[0028] Figure 13 is a PKI for web server certificate applications according to the present invention.

[0029] Figure 14 is a flow diagram illustrating a protocol for web server certificate applications for the PKI of Figure 13.

[0030] Figure 15 is a block diagram of an enterprise PKI according to the present invention.

[0031] Figure 16 is a flow diagram illustrating an authorization protocol for the enterprise PKI of Figure 17.

[0032] Figure 17 is a block diagram of a computer sys-

tem and a corresponding computer readable medium incorporating one or more main software program components of a PKI according to the present invention.

[0033] In the following detailed description of the preferred embodiments, reference is made to the accompanying drawings which form a part hereof, and in which is shown by way of illustration specific embodiments in which the invention may be practiced. It is to be understood that other embodiments may be utilized and structural or logical changes may be made without departing from the scope of the present invention. The following detailed description, therefore, is not to be taken in a limiting sense, and the scope of the present invention is defined by the appended claims.

[0034] A light-weight public key infrastructure (PKI) according to the present invention is illustrated generally at 30 in Figure 1. PKI 30 includes several main components which are each a software program. The main software program components of PKI 30 run on one or more computer systems. In one embodiment, each of the main software program components runs on its own computer system.

[0035] A certificate authority 32 issues short-term disposable certificates to one or more subjects, such as subject 34. Subject 34 is a user which owns a public key contained in the short-term disposable certificate and uses the short-term disposable certificate to demonstrate the subject's identity to one or more verifiers, such as verifier 36, by demonstrating that the subject has knowledge of a private key corresponding to the public key in the short-term disposable certificate. Verifier 36 is a user which relies on the short-term disposable certificate presented by subject 34 to verify that subject 34 is the subject of the short-term disposable certificate and that the attributes contained in the short-term disposable certificate apply to subject 34. In some embodiments of PKI 30, the same user can play the role of subject 34 and verifier 36 in different situations.

[0036] PKI 30 does not issue traditional long-term certificates, such as issued by conventional PKIs. Traditional long-term certificates typically have a validity period defined by an expiration date, which is a substantial period from the issue date, such as one year from the issue date. A conventional certificate authority needs to revoke a traditional long-term certificate if the public key certificate binding becomes invalid before the expiration date. As discussed in the Background of the Invention section of the present specification, a variety of methods have been used to revoke traditional long-term certificates, such as by using a certificate revocation list (CRL) or an on-line certificate status protocol (OCSP).

[0037] By contrast to traditional long-term certificates, the short-term disposable certificates issued by certificate authority 32 according to the present invention are not subject to revocation. The short-term disposable certificates are referred to as being disposable, because subject 34 can throw them away after a few uses. As a result, in one embodiment of the present invention, the short-term disposable certificates are stored in volatile memory and are not saved to a disk. PKI 30 is referred to as being a lightweight PKI because it is substantially simpler and more efficient than a conventional PKI.

[0038] A short-term disposable certificate is herein defined as a public key certificate that binds the subject's public key to one or more names or identifiers and has a short validity period determined by an expiration date/time. An example validity period for a short-term disposable certificate is one day, a few hours, or even a few minutes. Since the validity period is quite short, revocation is not necessary. Subject 34 requests a fresh short-term disposable certificate as needed and submits it to verifier 36. Verifier 36 only needs to check the expiration date/time to verify that the short-term disposable certificate is valid.

[0039] From the point of view of verifier 36, the use of short-term disposable certificates is a much better scheme than using traditional long-term certificates, because verifier 36 only has to verify a signature of the short-term disposable certificate and check the expiration date/time of the short-term disposable certificate. Since the certificate validation work of verifier 36 is likely to be a critical path in the overall work of verifier 36, short-term disposable certificates are a good solution for the verifier. In many cases, the verifier is a performance bottleneck and thus short-term disposable certificates are a good overall solution for PKI 30.

[0040] Certificate authority 32 includes a registration authority 38 that maintains a certificate database (DB) 40 containing unsigned certificates. Certificate authority 32 also includes a credentials server 42 that maintains a hash table (HT) 44 containing cryptographic hashes of unsigned certificates.

[0041] Registration authority 38 is an off-line component of certificate authority 32 and is responsible for subject registration and for maintaining certificate database 40. Each unsigned certificate in certificate database 40 binds a public key of a subject to one or more attributes that uniquely identify the subject.

[0042] Credentials server 42 is an on-line component of certificate authority 32 and is responsible for issuing the short-term disposable certificates and for maintaining the list of cryptographic hashes of currently valid unsigned certificates in hash table 44. Each cryptographic hash in hash table 44 is computed from an unsigned certificate using an agreed upon collision-resistant hash function, such as SHA-1 or MD5. Hash table 44 is essentially a list of the currently valid unsigned certificates which is keyed by the cryptographic hash. Cryptographic hashes function well as keys for hash table 44, because cryptographic hashes behave statically as random quantities.

[0043] Subject 34 has a private-key 46 stored in a secure storage medium 48, such as a smartcard or secure software wallet. Subject 34 also has a public key mathematically associated with private key 46. Subject 34 registers the public key corresponding to private key 46

with registration authority 38 by sending the public key and one or more attributes that uniquely identify subject 34's identity to registration authority 38 and demonstrating that the identification attributes apply to subject 34. Examples of such identification attributes include name, social security number, and employee number.

[0044] The components of PKI 30 are linked by one or more computer networks. A network connection 50 couples credentials server 42 and subject 34. A network connection 52 couples subject 34 and verifier 36. Network connections 50 and 52 are shown as solid lines in Figure 1 and are used for on-line communications. A network connection 54 couples registration authority 38 and subject 34. A network connection 56 couples registration authority 38 and credentials server 42. Network connections 54 and 56 are represented as dotted lines in Figure 1 and are used for off-line communications. Registration authority 38 is only involved in off-line communications and communicates through off-line network connections 54 and 56. On the other hand, credentials server 42 is involved in on-line communications through network connection 50.

[0045] One embodiment of an unsigned certificate issued by registration authority 38 is illustrated generally at 60 in Figure 2. Unsigned certificate 60 includes a meta-data (MD) field 61 containing data about unsigned certificate 60 itself rather than data related to the subject. Examples of data stored in meta-data field 61 include serial number and issuer name. Unsigned certificate 60 includes a subject's public key (PK) 62. Unsigned certificate 60 includes long-term identification information (LTI) field 63 containing attributes uniquely identifying subject 34, such as the subject's name, the subject's social security number, or the subject's employee number.

[0046] Unsigned certificate 60 optionally includes a long-term expiration (EXP) field 64 which contains a date/time of expiration for unsigned certificate 60. The expiration date/time contained in long-term expiration field 64 is useful for administrative purposes, but is not required for proper functioning of PKI 30. By contrast, in a conventional PKI, the expiration date is required to reduce the size of the CRL as revoked certificates reach their expiration dates. Unsigned certificate 60 optionally includes a duration (DUR) field 65 that specifies a duration for the validity period of any short-term disposable certificates issued against unsigned certificate 60.

[0047] An off-line protocol for issuing unsigned certificate 60 from registration authority 38 is illustrated generally at 100 in Figure 3. At step 102, subject 34 sends its public key and one or more attributes that uniquely identify subject 34 to registration authority 38.

[0048] At step 103, subject 34 demonstrates knowledge of the private key 46 associated with the subject 34's public key. Step 103, is performed in a way that depends on the cryptosystem for which the private-public key pair has been generated by subject 34. For example, in a digital signature cryptosystem, subject 34 demonstrates knowledge of the private key 46 by using private key 46 to digitally sign a quantity derived from a random quantity generated by registration authority 38. Registration authority 38 then verifies this digital signature using subject 34's public key.

[0049] At step 104, subject 34 demonstrates to registration authority 38, by out-of-band administrative means, that the identification attributes sent in step -102 apply to subject 34.

[0050] At step 106, registration authority 38 creates unsigned certificate 60 and stores unsigned certificate 60 in certificate database 40. At step 108, registration authority 38 sends unsigned certificate 60 to subject 34.

[0051] At step 110, registration authority 38 computes a cryptographic hash of unsigned certificate 60 using an agreed upon collision-resistant hash function, such as SHA-1 or MD5. In step 110, registration authority 38 sends the computed cryptographic hash of unsigned certificate 60 to credential server 42 over network connection 56, which provides data integrity protection, such as with a Secure Sockets Layer (SSL) connection.

[0052] At step 112, credentials server 42 stores the cryptographic hash of unsigned certificate 60 computed in step 110 in hash table 44.

[0053] One embodiment of a short-term disposable certificate as issued by credentials server 42 is illustrated generally at 70 in Figure 4. Short-term disposable certificate 70 includes a meta-data (MD) field 71 containing data about short-term disposable certificate 70 rather than the subject of short-term disposable certificate 70. Short-term disposable certificate 70 includes a public key (PK) 72 which is the same public key as public key 62 of unsigned certificate 60. The subject of unsigned certificate 60 and short-term disposable certificate 70 has only one private-public key pair associated with private key 46 stored in smartcard or secure software wallet 48. Short-term disposable certificate 70 includes long-term identification information (LTI) field 73 containing attributes uniquely identifying subject 34. Long-term identification information field 73 is identical to long-term identification information field 63 of unsigned certificate 60.

[0054] Short-term disposable certificate 70 also includes a short-term expiration (EXP) field 76 that specifies a date and time of expiration for a short-term disposable certificate 70. In one embodiment where unsigned certificate 60 contains a duration field 65 that specifies a duration for the validity period of the short-term disposable certificates issued therefrom, the date and time specified by short-term expiration field 76 is obtained by adding the duration specified by duration field 65 to the date and time at which short-term disposable certificate 70 is issued by credentials server 42. In one embodiment, the duration of the validity period of short-term disposable certificate 70 is not explicitly specified by a duration field 65 in unsigned certificate 60. In this embodiment, the duration of the validity period of short-term disposable certificate 70 is set by a policy.

[0055] Short-term disposable certificate 70 also includes a signature field 77 containing a signature of credentials server 42. The signature in signature field 77 is made by credentials server 42 on the sequence of fields 71, 72, 73, and 74, as well as, if necessary, the specification of the cryptosystem that has been used to produce the signature and must be used to verify the signature.

[0056] In one embodiment, short-term disposable certificate 70 is implemented with an X.509v3 certificate.

[0057] An on-line protocol for issuing a short-term disposable certificate 70 to subject 34 from credentials server 42 against unsigned certificate 60 is generally illustrated at 200 in Figure 5. At step 202, subject 34 sends a message to credentials server 42 containing unsigned certificate 60 and requesting that a short-term disposable certificate be issued against unsigned certificate 60.

[0058] At step 204, credentials server 42 computes a cryptographic hash of unsigned certificate 60 by the agreed upon collision-resistant hash function. In step 204, credentials server 42 then verifies that the computed cryptographic hash is present in hash table 44. At step 206, credentials server 42 creates short-term disposable certificate 70 and sends short-term disposable certificate 70 to subject 34.

[0059] In step 204, credentials server 42 performs the hash table 44 look-up with an efficient and computationally inexpensive operation. The signature operation performed by credentials server 42 in step 206 to create short-term disposable certificate 70, however, is a computationally expensive operation. Nevertheless, step 206 with the computationally expensive signature operation is only performed if the look-up of hash table 44 succeeds. The impact of a denial-of-service attack against credentials server 42 is reduced by only performing step 206 if the look-up in step 204 succeeds.

[0060] An on-line protocol employed by subject 34 to demonstrate its identity to verifier 36 is illustrated generally at 300 in Figure 6. At step 302, subject 34 sends the issued short-term disposable certificate 70 to verifier 36. At step 304, verifier 36 verifies that the expiration date and time specified in short-term expiration field 76 of short-term disposable certificate 70 has not been reached.

[0061] At step 306, verifier 36 uses a public key of credentials server 42 to verify the signature in signature field 77 of short-term disposable certificate 70. Verifier 36 knows the public key of credentials server 42 either directly or through certification by a higher certificate authority.

[0062] At step 308, subject 34 demonstrates knowledge of the private key 46 associated with the public key 72 contained in short-term disposable certificate 70. Step 308 is performed in a way that depends on the cryptosystem for which the private/public key pair has been generated by subject 34. For example, in a digital signature cryptosystem, subject 34 demonstrates knowledge of the private key 46 by using private key 46 to digitally sign a quantity derived from a random quantity generated by verifier 36. Verifier 36 then verifies this digital signature using the public key 72 in short-term disposable certificate 70.

[0063] Subject 34 can delete short-term disposable certificate 70 when the short-term disposable certificate expires, because subject 34 can obtain a new short-term disposable certificate by sending the unsigned certificate to credentials server 42. In one embodiment subject 34 stores short-term disposable certificate in volatile memory and does not save it to disk.

[0064] An off-line protocol for revoking unsigned certificate 60 is illustrated generally at 400 in Figure 7. Off-line protocol 400 is performed when subject 34's private key 46 is compromised or the identification attributes in long-term identification information field 63 no longer apply to subject 34, because in either case the binding of public key 62 to the identification attributes is invalid.

[0065] At step 402, registration authority 38 retrieves unsigned certificate 60 from certificate database 40 and computes a cryptographic hash of unsigned certificate 60 using the agreed upon collision-resistant hash function. In one embodiment, registration authority 38 marks unsigned certificate 60 in certificate database 40 as being invalid, for auditing purposes. In an alternative embodiment, where retention of unsigned certificate 60 in certificate database 40 is not required for auditing purposes, registration authority 38 deletes unsigned certificate 60 from certificate database 40.

[0066] At step 404, registration authority 38 sends a message to credentials server 42 containing the cryptographic hash of unsigned certificate 60 computed in step 402. The message sent in step 404 also requests that credentials server 42 remove the corresponding cryptographic hash of unsigned certificate 60 from hash table 44.

[0067] At step 406, credentials server 42 removes the cryptographic hash of unsigned certificate 60 from hash table 44 which corresponds to the cryptographic hash sent in the message from registration authority 38 in step 404. After step 406 is completed, credentials server 42 no longer issues short-term disposable certificates 70 against unsigned certificate 60. Consequently, once protocol 400 has been executed, neither subject 34 nor an attacker can obtain a short-term disposable certificate by presenting unsigned certificate 60 to credentials server 42.

[0068] An alternate embodiment light-weight PKI according to the present invention is illustrated generally at 30' in Figure 8. The PKI 30' components 32', 34', 36', 38', 40', 42', 44', 46', 48', 50', 52', 54', and 56' generally operate and are generally coupled substantially the same as the corresponding PKI 30 components 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, and 56 described above.

[0069] PKI 30', however, includes a directory 90. In one embodiment, directory 90 is a light-weight directory

access protocol (LDAP) directory. PKI 30' also includes a network connection 92 to couple credentials server 42' to directory 90 to permit credentials server 42' to access directory 90. Credentials server 42' obtains short-term authorization information stored in directory 90. Credentials server 42' adds the short-term authorization information obtained from directory 90 to short-term disposable certificates issued by credentials server 42', to create short-term credentials certificates which can be employed for authorization of subject 34'.

[0070] The short-term authorization information data contained in directory 90 relates to attribute or authorization information about subject 34'. Example short-term authorization information includes expense authorization limit for an enterprise application, co-payment amount for an insurance application, disk storage quota for an Information Technology (IT) application, and user ID plus Group ID for Unix access application.

[0071] In PKI 30', verifier 36' is an application program running on a server computer system and subject 34' is a client program that uses the application.

[0072] Unsigned certificates issued by registration authority 38' are substantially similar to the unsigned certificates issued by registration authority 38, such as unsigned certificate 60 of Figure 2.

[0073] PKI 30' employs off-line protocol 100 illustrated in Figure 3 for issuing unsigned certificate 60 from registration authority 38. In addition, PKI 30' employs off-line protocol 400 illustrated in Figure 7 for registration authority 38' to revoke unsigned certificate 60.

[0074] One embodiment of a structured short-term disposable certificate issued by credentials server 42' is illustrated generally at 80 in Figure 9. Short-term disposable certificate 80 is a structured certificate.

[0075] Structured short-term disposable certificate 80 includes a meta-data field (MD) field 81, a public key (PK) 82, a short-term expiration field 86, and a signature field 87, which are substantially similar to meta-data field 71, public key 72, short-term expiration field 76 and signature field 77 of non-structured short-term disposable certificate 70 of Figure 4. Structured short-term disposable certificate 80, however, includes folders 88a through 88n, which respectively correspond to applications 36'a through 36'n. For every verifier/application 36' that the subject/client 34' can access, as specified, for example, by a user profile, there is a cryptographic folder 88. Each cryptographic folder 88 contains long-term identification information 83 and/or short-term authorization information 89 as required by the corresponding verifier/application 36' to make authorization decisions about the subject/client 34'. In one embodiment, structured short-term disposable certificate 80 is implemented by adding a folder extension to an X.509v3 certificate.

[0076] An on-line protocol for issuing a structured short-term disposable certificate 80 to subject/client 34' from credentials server 42' against unsigned certificate 60 is generally illustrated at 500 in Figure 10. At step 502, subject/client 34' sends a message to credentials server 42' containing unsigned certificate 60 and requesting that a short-term disposable certificate be issued against unsigned certificate 60.

[0077] At step 504, credentials server 42' computes a cryptographic hash of unsigned certificate 60 with an agreed upon collision-resistant hash function. In step 504, credentials server 42' then verifies that the computed cryptographic hash is present in hash table 44'.

[0078] At step 506, credentials server 42' accesses directory 90 via network connection 92 and obtains short-term authorization information for structured short-term disposable certificate 80.

[0079] At step 508, credentials server 42' combines the short-term authorization information obtained from directory 90 in step 506 with the identification attributes in long-term identification information field 63 of unsigned certificate 60. In step 508, credentials server 42' creates a cryptographic folder 88 for each verifier/application 36' that can be accessed by subject/client 34', where each cryptographic folder 88 contains all the long-term identification information and/or short-term authorization information required by verifier/application 36' to make authorization decisions about subject/client 34'. In step 508, credentials server 42' uses the cryptographic folders 88 to create structured short-term disposable certificate 80.

[0080] At step 510, credentials server 42' sends structured short-term disposable certificate 80 to subject/client 34', with all of the short-term disposable certificate's folders open.

[0081] An on-line protocol for authorizing subject/client 34' is illustrated generally at 600 in Figure 11. On-line protocol 600 is employed by subject/client 34' to demonstrate its identity to verifier/application 36'. On-line protocol 600 is also used by verifier/application 36' to make authorization decisions concerning subject/client 34', such as allowing/denying access or authorizing specific transactions.

[0082] At step 602, subject/client 34' closes all folders 88 in structured short-term disposable certificate 80, except the folder that contains the necessary identification/authorization information 83/89 required by verifier/application 36' to make authorization decisions concerning subject/client 34'. At step 604, subject/client 34' sends structured short-term disposable certificate 80 to verifier/application 36'.

[0083] At step 606, verifier/application 36' verifies that the expiration date/time specified in expiration field 86 of structured short-term disposable certificate 80 has not expired.

[0084] At step 608, verifier/application 36' uses a public key of credentials server 42' to verify the signature in signature field 87 of structured short-term disposable certificate 80. Verifier/application 36' knows the public key of credentials server 42' either directly or through certification by a higher certificate authority.

[0085] At step 610, subject/client 34' demonstrates

knowledge of the private key 46' associated with the public key 82 of structured short-term disposable certificate 80. Step 610 is performed in a way that depends on the cryptosystem for which the private/public key pair has been generated by subject/client 34'. For example, in a digital signature cryptosystem, subject/client 34' demonstrates knowledge of the private key 46' by using private key 46' to digitally sign a quantity derived from a random quantity generated by verifier/application 36'. Verifier/application 36' then verifies this digital signature using the public key 82 of structured short-term disposable certificate 80'.

[0086] At step 612, verifier/application 36' extracts the identification/authorization information 83/89 contained in the open folder 88 of structured short-term disposable certificate 80. In step 612, verifier/application 36' then uses the identification/authorization information 83/89 to make authorization decisions concerning subject/client 34'.

[0087] On-line protocol 600 for authorizing subject/client 34' utilizes structured short-term disposable certificate 80 having one folder for each application that may be accessed by subject/client 34' as determined by a user profile. This ensures that each application only has access to authorization information that it requires. Nevertheless, the authorization performed by PKI 30' can be implemented with non-structured short-term disposable certificates, where multiple non-structured short-term disposable certificates are employed in place of one structured short-term disposable certificate 80.

[0088] The PKI 30/30' according to the present invention is greatly simplified and substantially more efficient than conventional PKIs. For example, applications only need to use the short-term disposable certificate for authentication and/or authorization. The unsigned certificate 60, which replaces the traditional long-term certificate, can be reserved for use by the subject when requesting a short-term disposable certificate. Since the unsigned certificate 60 is not used by applications, it does not need to be signed. Instead of signing the unsigned certificate 60, the credentials server keeps the cryptographic hash table 44, which contains the cryptographic hashes of the unsigned certificates that are currently valid. In this way, certificate revocation is performed simply by removing the cryptographic hash of the unsigned certificate from hash table 44. Therefore, unlike conventional PKIs, there is no signature required, no expiration date required, and no need for CRLs for unsigned certificate 60 of the PKI 30/30' according to the present invention.

[0089] PKI 30/30' requires a certificate status check somewhat like the on-line certificate status check required by OCSP. The certificate status check of PKI 30/30' according to the present invention, however, occurs when the subject requests the short-term disposable certificate, rather than when the subject accesses the application, such as required by OCSP.

[0090] A distributed certificate authority 132 for use in PKI 30/30' according to the present invention is illustrated in Figure 12. Distributed certificate authority 132 includes a registration authority 138 having a certificate database 140 which communicates with a distributed credentials server 142.

[0091] Distributed credentials server 142 includes credentials servers 142a through 142n. Each credentials server 142 includes a corresponding hash table partition 144. In one embodiment, cryptographic hash table 144 is partitioned into hash table partitions 144a through 144n according to a value of some of the bits of the cryptographic hash. In a PKI 30/30' employing distributed certificate authority 132 having distributed credentials servers 142a-142n, the subject sends a request for a short-term disposable certificate directly to the correct hash table partition 144.

[0092] Certificate authority 132 is further optimized by directly coupling each credentials server 142 with its own replica of directory 190.

[0093] The PKI 30/30' according to the present invention is highly scalable because many bottlenecks of conventional PKIs are removed. The unsigned certificates 60 employed by PKI 30/30' do not expire and do not have to be renewed periodically. In addition, CRLs are not used. No significant bottlenecks have been introduced with PKI 30/30'.

[0094] One potential bottleneck of PKI 30/30' is that the credentials server 42/42'/142 issues short-term disposable certificates very often, such as once a day, every few hours, or even every few minutes. This frequency of issuing short-term disposable certificates from the credentials server according to the present invention, however, is not a significant bottleneck, because the hash table can be partitioned, such as hash table portions 144a-144n of the distributed certificate authority 132 illustrated in Figure 12 and the credentials server can be replicated as required, such as credentials servers 142a-142n of distributed certificate authority 132.

[0095] Another potential bottleneck with PKI 30/30' is the certificate database 40/40' of registration authority 38/38'. However, certificate database 40/40' is not a significant bottleneck, because the certificate database is only accessed when unsigned certificates 60 are issued or revoked. Certificate database 40/40' can be implemented using a relational database management system (DBMS), which can handle millions of unsigned certificates.

[0096] Another potential bottleneck with PKI 30/30' is directory 90. Directory 90 is not a significant bottleneck, because directory 90 can be replicated to make it possible to handle millions of users. In addition, LDAP traffic to directory 90 is significantly reduced, since applications do not access directory 90 to make authorizations decisions, since all necessary authorization information is contained in short-term disposable certificate 80 in the embodiment where PKI 30' is used for authorizing the subject. Therefore, PKI according to the present invention can scale to millions of users.

Example Applications of PKI According to the Present Invention

[0097] The PKI according to the present invention can be used advantageously in a broad range of applications. The following two examples provide only two of numerous applications for the PKI according to the present invention.

Web Server Certificates

[0098] A PKI according to the present invention for a web server certificate application is illustrated generally at 730 in Figure 13. PKI 730 includes a certificate authority 732 (with credentials server), a web server/subject 734, and a browser/verifier 736, which are all coupled through an Internet 731.

[0099] PKI 730 does not use short-term authorization information in its short-term disposable certificate. Therefore, a suitable short-term disposable certificate for PKI 30 is short-term disposable certificate 70 illustrated in Figure 4, which is a non-structured certificate (i.e., no folders). In addition, the short-term disposable certificate 70 employed in PKI 730 contains no confidential information.

[0100] A web server certificate protocol for PKI 730 is illustrated generally at 700 in Figure 14. At step 702, certificate authority 732 issues an unsigned certificate 60 for web server 734's public key. The unsigned certificate is not signed, is retrieved only once, and never expires.

[0101] At step 704, web server 734 requests a non-structured short-term disposable certificate 70. The unsigned certificate 60 is sent as part of the web server request. In one embodiment, meta-data field 61 specifies the duration of the validity period of the to be issued short-term disposable certificate 70.

[0102] At step 706, certificate authority 732 verifies the unsigned certificate 60 provided by web server 734 and issues a corresponding short-term disposable certificate 70. In step 706, short-term disposable certificate 70 is derived from the unsigned certificate 60 by adding an expiration time in short-term expiration field 76, such as 24 hours after the present time, and by adding the certificate authority 732's signature in signature field 77.

[0103] At step 708, web server 734 sends short-term disposable certificate 70 to browser 736. Step 708 is part of a protocol for establishing an SSL connection. At step 710, a handshake is performed between browser 736 and web server 734. In step 710, web server 734 demonstrates to browser 736 that web server 734 knows the private key corresponding to the public key in short-term disposable certificate 70. Step 710 is also part of the protocol for establishing the SSL connection.

[0104] At step 712 a secure connection is made between browser 736 and web server 734. In step 712, an example secure connection is SSL which provides data integrity, which prevents connection high-jacking and other man-in-the-middle attacks. SSL also provides encryption to ensure confidentiality.

[0105] In conventional web server certificate PKI applications, the web server's certificate can not be revoked or the browser must access the certificate authority to obtain the latest CRL or check the certificate status via OCSP. Having the browser of a conventional web server certificate PKI access the certificate authority is highly undesirable, because this access to the certificate authority introduces a delay which is perceived by the user as additional delay in accessing the web server. In addition, this browser access of the certificate authority possibly prevents access to the web server if the certificate authority is down. Moreover, CRL or OCSP processing requires additional code in the browser, which may not fit in the browser if the browser runs on a small network appliance.

[0106] By contrast to the conventional PKI, in PKI 730 according to the present invention, certificate authority 732 can revoke the web server's unsigned certificate 60. After certificate authority 732 has revoked unsigned certificate 60, web server 734 is no longer able to obtain short-term disposable certificates, and browsers 736 are not able to establish SSL connections to web server 734. Moreover, browser 736 does not use CRLs or OCSP, so browser access of certificate authority 732 to obtain the latest CRL or check the certificate status via OCSP is not required with PKI 730.

Enterprise PKI

[0107] An enterprise PKI according to the present invention is illustrated generally at 930 in Figure 15. Enterprise PKI 930 includes a certificate authority 932 (with credentials server) coupled to a client 934 via a network connection 950. Certificate authority 932 is coupled to a LDAP directory 990 via a network connection 992. Client 934 has access to a user's private key 946 stored in a smartcard 948. Client 934 is coupled to applications 936a, 936b, and 936c via network connections 952a, 952b, and 952c respectively.

[0108] The network connections of enterprise PKI 930 are deemed secure or are protected by host-to-host IPSEC. Client 934's short-term disposable certificate is a structured certificate, such as structured short-term disposable certificate 80 illustrated in Figure 9. Structured short-term disposable certificate 80 in enterprise PKI 930 contains short-term authorization information that is possibly confidential.

[0109] An authorization protocol for enterprise PKI 930 is illustrated generally at 900 in Figure 16. At step 902, certificate authority 932 issues an unsigned certificate 60 for the user's public key. Step 902 needs only be taken once. At step 904, client 934 submits unsigned certificate 60 to certificate authority 932. Step 904 is taken at network logon.

[0110] At step 906, a handshake between certificate authority 932 and client 934 permits client 934 to dem-

onstrate to certificate authority 932 that client 934 has access to the user's private key 946 stored in smartcard 948. Step 906 is taken at network logon. At step 908, certificate authority 932 obtains short-term authorization information from LDAP directory 990. In one embodiment, LDAP directory 990 is integrated with certificate authority 932. Step 908 is taken at network logon.

[0111] At step 910, certificate authority 932 issues structured short-term disposable certificate 80. In step 910, certificate authority 932 sends the issued structured short-term disposable certificate 80 to client 934. In an example application of enterprise PKI 930, the structured short-term disposable certificate 80 includes three folders, one for each application 936a, 936b, and 936c. All of the folders of structured short-term disposable certificate 80 are open. Some of the information in the folders of structured short-term disposable certificate 80 is possibly confidential. Network connections 952, however, are deemed secure or are protected by host-to-host IPSEC. Step 910 is taken at network logon.

[0112] At step 912, client 934 submits structured short-term disposable certificate 80 to the requested application 936. For example, if the requested application is 936b, the folder 88b, corresponding to application 936b, remains open. In this example, the folders 88a and 88c of structured short-term disposable certificate 80, which correspond to applications 936a and 936c, are closed.

[0113] At step 914, a handshake is performed between the requested application 936b and client 934. In step 914, client 934 demonstrates to application 936b that client 934 has access to the user's private key 946 stored in smartcard 948.

[0114] The enterprise PKI 930 according to the present invention is streamlined compared to conventional enterprise PKIs. Enterprise PKI 930 does not need to use CRLs or OCSP. All information needed to make authorizations decisions is contained in structured short-term disposable certificate 80. Therefore, applications 936 do not need to access LDAP directory 990 to make authorizations decisions about client 934.

[0115] Enterprise PKI 930 can handle millions of users. PKI 930 is suitable to provide authentication and authorization services for all employees and business partners of a corporation of any size.

[0116] Figure 17 illustrates one embodiment of a computer system 250 and an external computer readable medium 252 which can be employed according to the present invention to implement one or more of the main software program components of a light-weight PKI according to the present invention, such as PKI 30, PKI 30', PKI 730, PKI 830 and PKI 930. Embodiments of external computer readable medium 252 include, but are not limited to: a CD-ROM, a floppy disk, and a disk cartridge. Any one of the main software program components of a light-weight PKI according to the present invention can be implemented in a variety of compiled and interpreted computer languages. External computer readable medium 252 stores source code, object code, executable code, shell scripts and/or dynamic link libraries for any one of the main software program components of a light-weight PKI according to the present invention. An input device 254 reads external computer readable medium 252 and provides this data to computer system 250. Embodiments of input device 254 include but are not limited to: a CD-ROM reader, a floppy disk drive, and a data cartridge reader.

[0117] Computer system 250 includes a central processing unit 256 for executing any one of the main software program components of a light-weight PKI according to the present invention. Computer system 250 also includes local disk storage 262 for locally storing any one of the main software program components of a light-weight PKI according to the present invention before, during, and after execution. Any one of the main software program components of a light-weight PKI according to the present invention also utilizes memory 260 within the computer system during execution. Upon execution of any one of the main software program components of a light-weight PKI according to the present invention, output data is produced and directed to an output device 258. Embodiments of output device 258 include, but are not limited to: a computer display device, a printer, and/or a disk storage device.

[0118] Although specific embodiments have been illustrated and described herein for purposes of description of the preferred embodiment, it will be appreciated by those of ordinary skill in the art that a wide variety of alternate and/or equivalent implementations calculated to achieve the same purposes may be substituted for the specific embodiments shown and described without departing from the scope of the present invention. Those with skill in the chemical, mechanical, electromechanical, electrical, and computer arts will readily appreciate that the present invention may be implemented in a very wide variety of embodiments. This application is intended to cover any adaptations or variations of the preferred embodiments discussed herein. Therefore, it is manifestly intended that this invention be limited only by the claims and the equivalents thereof.

Claims

1. A public key infrastructure (PKI) (30) comprising:

   a subject (34);

   an off-line registration authority (38) for issuing a first unsigned certificate (60) off-line to the subject that binds a public key (62) of the subject to long-term identification information (63) related to the subject, the registration authority maintaining a certificate database (40) of unsigned certificates in which it stores the first unsigned certificate;

   an on-line credentials server (42) for issuing a

short-term disposable certificate (70) on-line to the subject, the short-term disposable certificate binds the public key of the subject from the first unsigned certificate to the long-term identification information related to the subject from the first unsigned certificate, wherein the credentials server maintains a table (44) that contains entries corresponding to valid unsigned certificates stored in the certificate database; and

a verifier (36), wherein the subject presents the short-term disposable certificate to the verifier for authentication and demonstrates that the subject has knowledge of a private key (46) corresponding to the public key in the short-term disposable certificate.

2. The PKI of claim 1 wherein the short-term disposable certificate includes an expiration date/time.

3. The PKI of claim 2 wherein a validity period from when the credentials server issues the short-term disposable certificate to the expiration date/time is sufficiently short such that the short-term certificate does not need to be subject to revocation.

4. The PKI of claim 1 or 2 wherein the short-term disposable certificate is not subject to revocation.

5. The PKI of claim 1 wherein the table maintained by the credentials server is a hash table containing cryptographic hashes of valid unsigned certificates stored in the certificate database and including a cryptographic hash of the first unsigned certificate, wherein the subject presents the issued first unsigned certificate to the credentials server to obtain the short-term disposable certificate.

6. The PKI of claim 1 wherein the registration authority and the credentials server are included in a certificate authority and wherein the certificate authority is a distributed certificate authority including at least two distributed credentials servers.

7. The PKI of claim 6 wherein the at least two distributed credentials servers includes at least two corresponding hash table partitions containing cryptographic hashes of valid unsigned certificates corresponding to the unsigned certificates stored in the certificate database, wherein one of the hash table partitions contains a cryptographic hash of the first unsigned certificate, wherein the subject presents the issued first unsigned certificate to one of the at least two distributed credentials servers to obtain the short-term disposable certificate.

8. The PKI of claim 1 wherein the short-term disposable certificate is,a non-structured short-term certificate.

9. The PKI of claim 1 further comprising:

a directory for storing short-term authorization information related to the subject;

wherein the short-term disposable certificate also binds the public key of the subject from the first unsigned certificate to the short term authorization information related to the subject from the directory; and

wherein the subject presents the short-term disposable certificate to the verifier for authorization and demonstrates that the subject has knowledge of a private key corresponding to the public key in the short-term disposable certificate.

10. The PKI of claim 1 further comprising:

a second verifier; and

wherein the short-term certificate is a structured short-term certificate including:

a first folder corresponding to the first named verifier and containing long-term information and short-term information as required by the first named verifier;

a second folder corresponding to the second verifier and containing long-term information and short-term information as required by the second verifier;

wherein the first folder is open and the second folder is closed when the subject presents the short-term disposable certificate to the first named verifier for authorization, wherein closing the second folder makes its contents not readable by the first named verifier; and

wherein the first folder is closed and the second folder is open when the subject presents the short-term disposable certificate to the second verifier for authorization, wherein closing the first folder makes its contents not readable by the second verifier.

11. The PKI of claim 10 wherein the first folder and the second folder are implemented as extension fields of an X.509v3 certificate.

12. The PKI of claim 1 wherein the registration authority and the credentials server are included in a certificate authority and wherein the certificate authority revokes the first unsigned certificate when the binding of the subject's public key to the long-term identification information related to the subject becomes invalid.

13. The PKI of claim 19 wherein the certificate authority performs a revocation protocol to revoke the first unsigned certificate, the revocation protocol including:

the registration authority retrieving the first unsigned certificate from the certificate database and computing a cryptographic hash of the first unsigned certificate;
sending a message from the registration authority to credentials server containing the cryptographic hash of the first unsigned certificate and requesting that the credentials server remove the corresponding cryptographic hash of the first unsigned certificate from its hash table; and
the credentials server removing the cryptographic hash of the first unsigned certificate from its hash table.

14. A method of authenticating a subject (34), the method comprising:

issuing a first unsigned certificate (60) off-line to the subject that binds a public key (62) of the subject to long-term identification information (63) related to the subject;
maintaining a certificate database (40) of unsigned certificates off-line including storing the first unsigned certificate in the certificate database;
issuing a short-term disposable certificate (70) on-line to the subject, the short-term disposable certificate binds the public key of the subject from the first unsigned certificate to the long-term identification information related to the subject from the first unsigned certificate;
maintaining a table (44) on-line that contains entries corresponding to valid unsigned certificates stored in the certificate database; and
presenting the short-term disposable certificate by the subject to a verifier (36) for authentication and demonstrating that the subject has knowledge of a private key (46) corresponding to the public key in the short-term disposable certificate.

15. The method of claim 14 wherein the short-term disposable certificate includes an expiration date/time.

16. The method of claim 15 wherein a validity period from when the short-term disposable certificate is issued to the expiration date/time is sufficiently short such that the short-term certificate does not need to be subject to revocation.

17. The method of claim 14 or 15 wherein the short-term disposable certificate is not subject to revocation.

18. The method of claim 14 wherein the table is maintained by a credentials server and is a hash table containing cryptographic hashes of valid unsigned certificates stored in the certificate database and including a cryptographic hash of the first unsigned certificate, wherein the method further includes:
presenting the issued first unsigned certificate by the subject to the credentials server to obtain the short-term disposable certificate.

19. The method of claim 14 wherein the short-term disposable certificate is a non-structured short-term certificate.

20. The method of claim 14 further comprising:

storing short-term authorization information related to the subject in a directory, wherein the short-term disposable certificate also binds the public key of the subject from the first unsigned certificate to the short term authorization information related to the subject from the directory; and
presenting the short-term disposable certificate by the subject to the verifier for authorization and demonstrating that the subject has knowledge of a private key corresponding to the public key in the short-term disposable certificate.

21. The method of claim 14 wherein the short-term certificate is a structured short-term certificate including a first folder corresponding to the first named verifier and containing long-term information and short-term information as required by the first named verifier, and including a second folder corresponding to a second verifier and containing long-term information and short-term information as required by the second verifier, the method further comprising:

closing the second folder and leaving the first folder open prior to the presenting step if the short-term disposable certificate is presented by the subject to the first named verifier for authorization, wherein closing the second folder makes its contents not readable by the first named verifier; and
closing the first folder and leaving the second folder open prior to the presenting step if the short-term disposable certificate is presented by the subject to the second verifier for authorization, wherein closing the first folder makes its contents not readable by the second verifier.

22. The method of claim 14 further comprising:
revoking the first unsigned certificate when

the binding of the subject's public key to the long-term identification information related to the subject becomes invalid.

23. The method of claim 22 further comprising:

performing a revocation protocol to revoke the first unsigned certificate, the revocation protocol including:

retrieving the first unsigned certificate from the certificate database and computing a cryptographic hash of the first unsigned certificate;
sending a message containing the cryptographic hash of the first unsigned certificate request that the corresponding cryptographic hash of the first unsigned certificate be removed from its hash table; and
removing the cryptographic hash of the first unsigned certificate from its hash table.

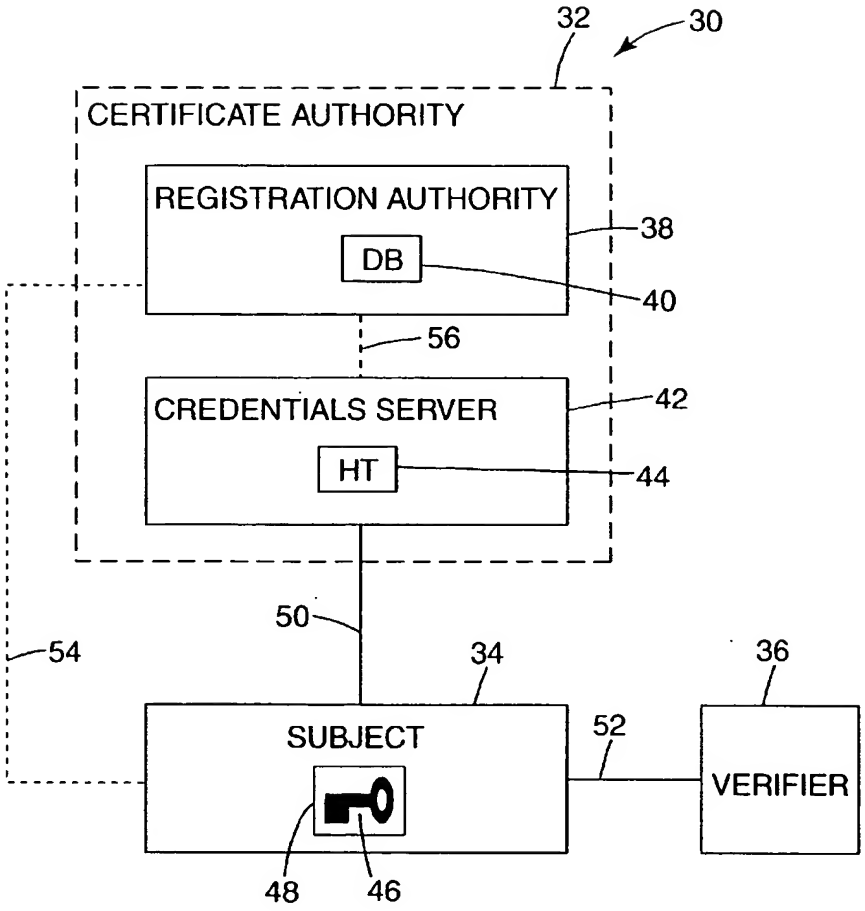
32
30
CERTIFICATE AUTHORITY
REGISTRATION AUTHORITY
38
DB
40
56
CREDENTIALS SERVER
42
HT
44
50
54
34
36
SUBJECT
52
VERIFIER
48
46

Fig. 1

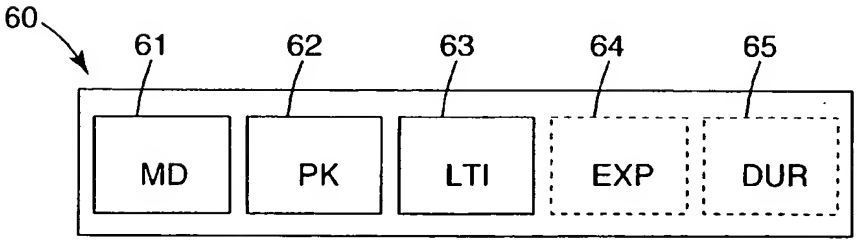
60
61
62
63
64
65
MD
PK
LTI
EXP
DUR

Fig. 2

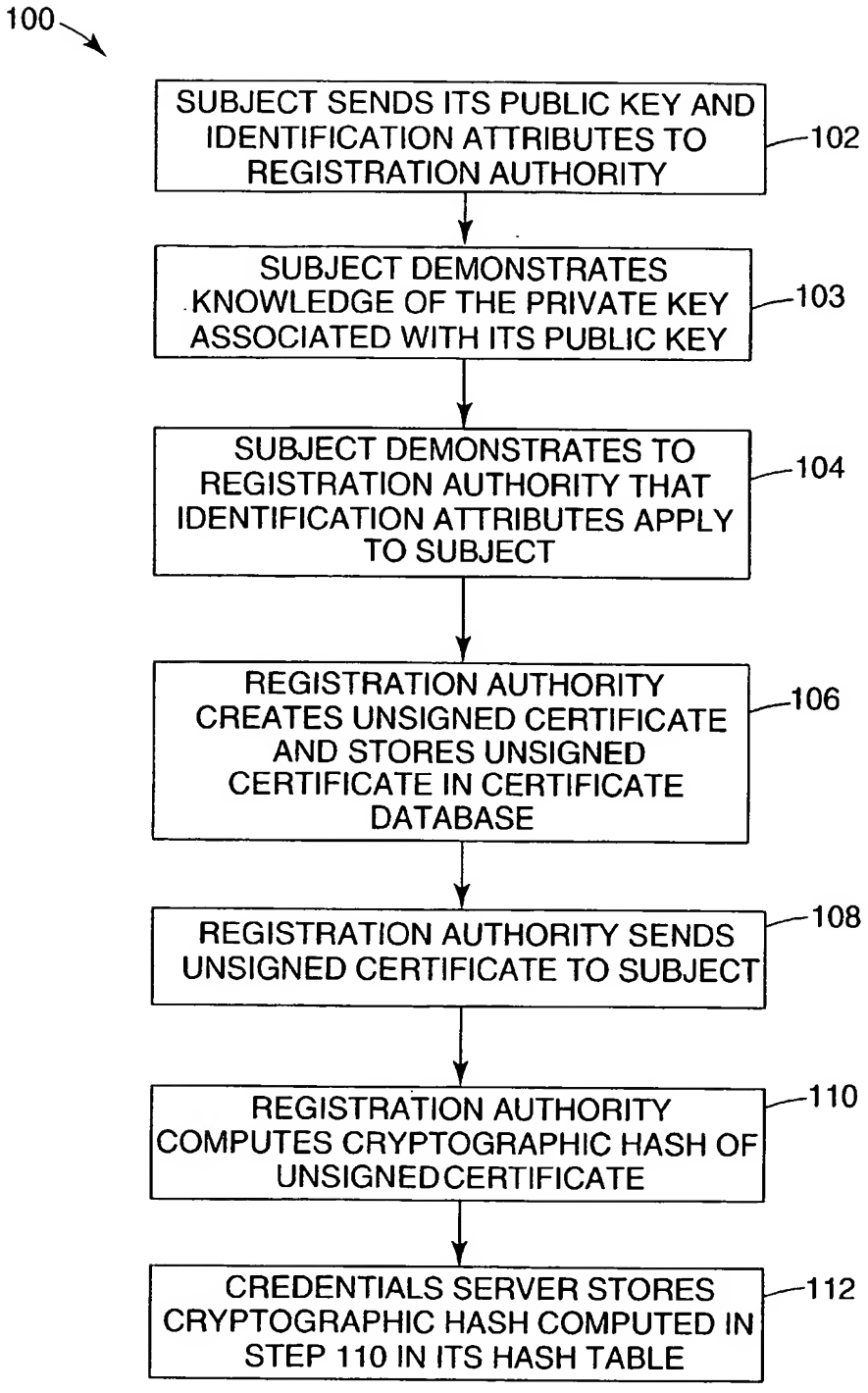
100
SUBJECT SENDS ITS PUBLIC KEY AND IDENTIFICATION ATTRIBUTES TO REGISTRATION AUTHORITY
102
SUBJECT DEMONSTRATES KNOWLEDGE OF THE PRIVATE KEY ASSOCIATED WITH ITS PUBLIC KEY
103
SUBJECT DEMONSTRATES TO REGISTRATION AUTHORITY THAT IDENTIFICATION ATTRIBUTES APPLY TO SUBJECT
104
REGISTRATION AUTHORITY CREATES UNSIGNED CERTIFICATE AND STORES UNSIGNED CERTIFICATE IN CERTIFICATE DATABASE
106
REGISTRATION AUTHORITY SENDS UNSIGNED CERTIFICATE TO SUBJECT
108
REGISTRATION AUTHORITY COMPUTES CRYPTOGRAPHIC HASH OF UNSIGNEDCERTIFICATE
110
CREDENTIALS SERVER STORES CRYPTOGRAPHIC HASH COMPUTED IN STEP 110 IN ITS HASH TABLE
112

Fig. 3

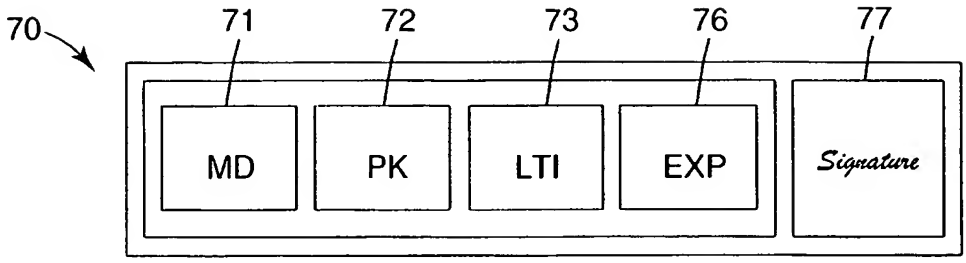
70
71
72
73
76
77
MD
PK
LTI
EXP
Signature

Fig. 4

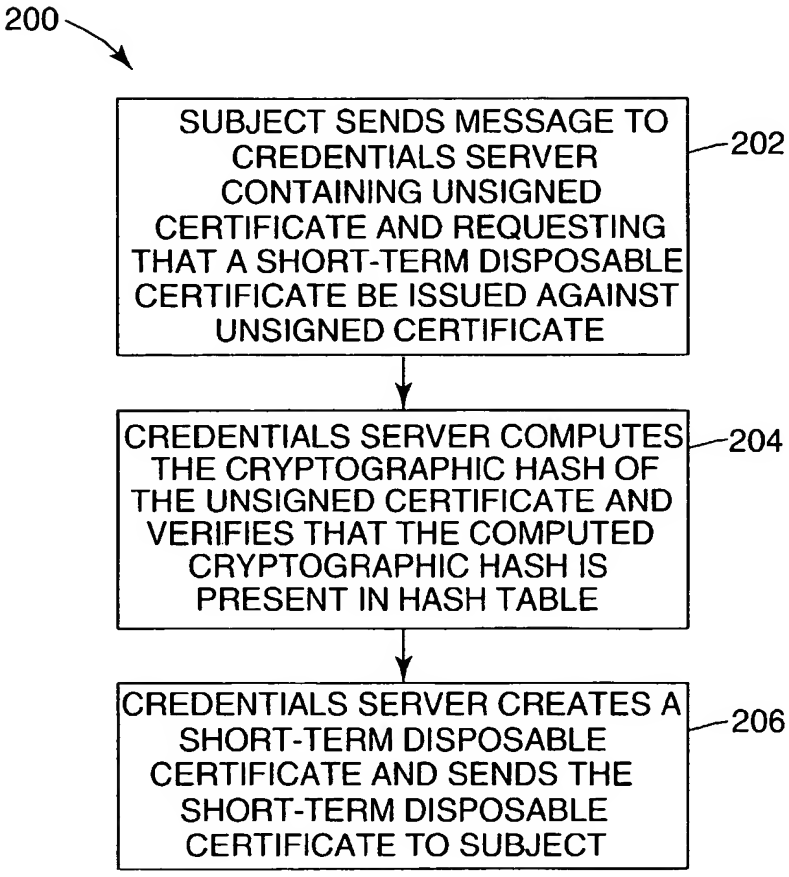
200
SUBJECT SENDS MESSAGE TO CREDENTIALS SERVER CONTAINING UNSIGNED CERTIFICATE AND REQUESTING THAT A SHORT-TERM DISPOSABLE CERTIFICATE BE ISSUED AGAINST UNSIGNED CERTIFICATE
202
CREDENTIALS SERVER COMPUTES THE CRYPTOGRAPHIC HASH OF THE UNSIGNED CERTIFICATE AND VERIFIES THAT THE COMPUTED CRYPTOGRAPHIC HASH IS PRESENT IN HASH TABLE
204
CREDENTIALS SERVER CREATES A SHORT-TERM DISPOSABLE CERTIFICATE AND SENDS THE SHORT-TERM DISPOSABLE CERTIFICATE TO SUBJECT
206

Fig. 5

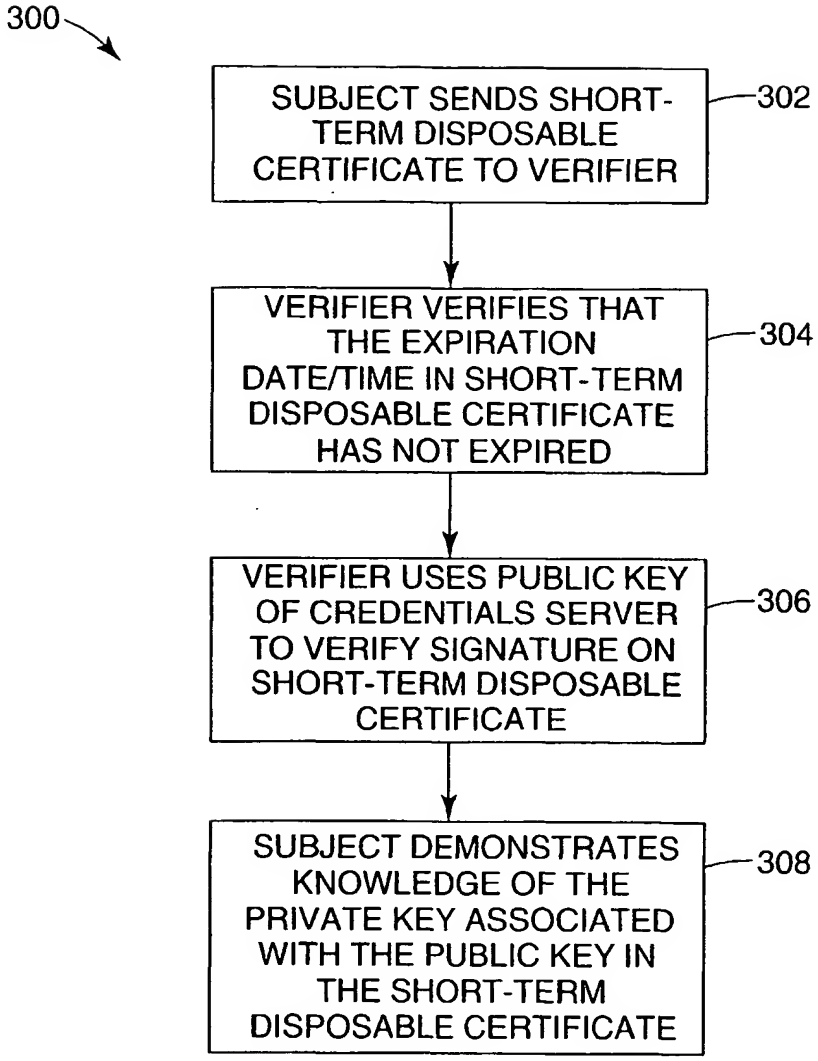
300
SUBJECT SENDS SHORT-TERM DISPOSABLE CERTIFICATE TO VERIFIER
302
VERIFIER VERIFIES THAT THE EXPIRATION DATE/TIME IN SHORT-TERM DISPOSABLE CERTIFICATE HAS NOT EXPIRED
304
VERIFIER USES PUBLIC KEY OF CREDENTIALS SERVER TO VERIFY SIGNATURE ON SHORT-TERM DISPOSABLE CERTIFICATE
306
SUBJECT DEMONSTRATES KNOWLEDGE OF THE PRIVATE KEY ASSOCIATED WITH THE PUBLIC KEY IN THE SHORT-TERM DISPOSABLE CERTIFICATE
308

Fig. 6

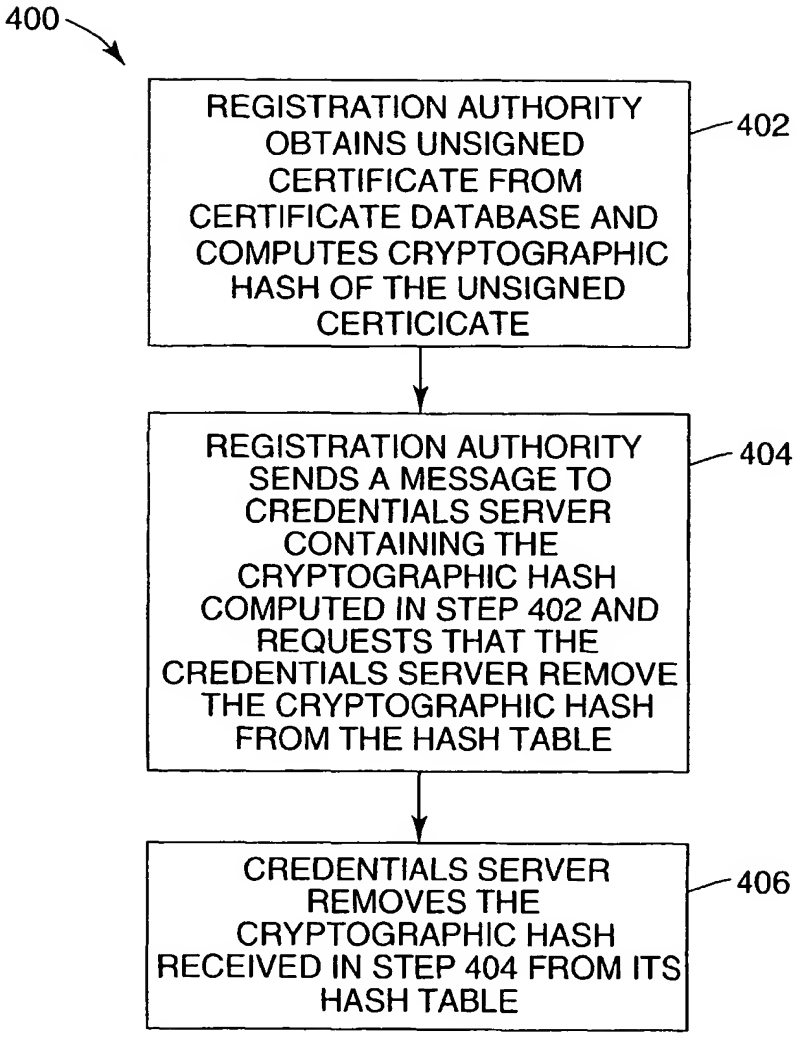
400
REGISTRATION AUTHORITY OBTAINS UNSIGNED CERTIFICATE FROM CERTIFICATE DATABASE AND COMPUTES CRYPTOGRAPHIC HASH OF THE UNSIGNED CERTICICATE
402
REGISTRATION AUTHORITY SENDS A MESSAGE TO CREDENTIALS SERVER CONTAINING THE CRYPTOGRAPHIC HASH COMPUTED IN STEP 402 AND REQUESTS THAT THE CREDENTIALS SERVER REMOVE THE CRYPTOGRAPHIC HASH FROM THE HASH TABLE
404
CREDENTIALS SERVER REMOVES THE CRYPTOGRAPHIC HASH RECEIVED IN STEP 404 FROM ITS HASH TABLE
406

Fig. 7

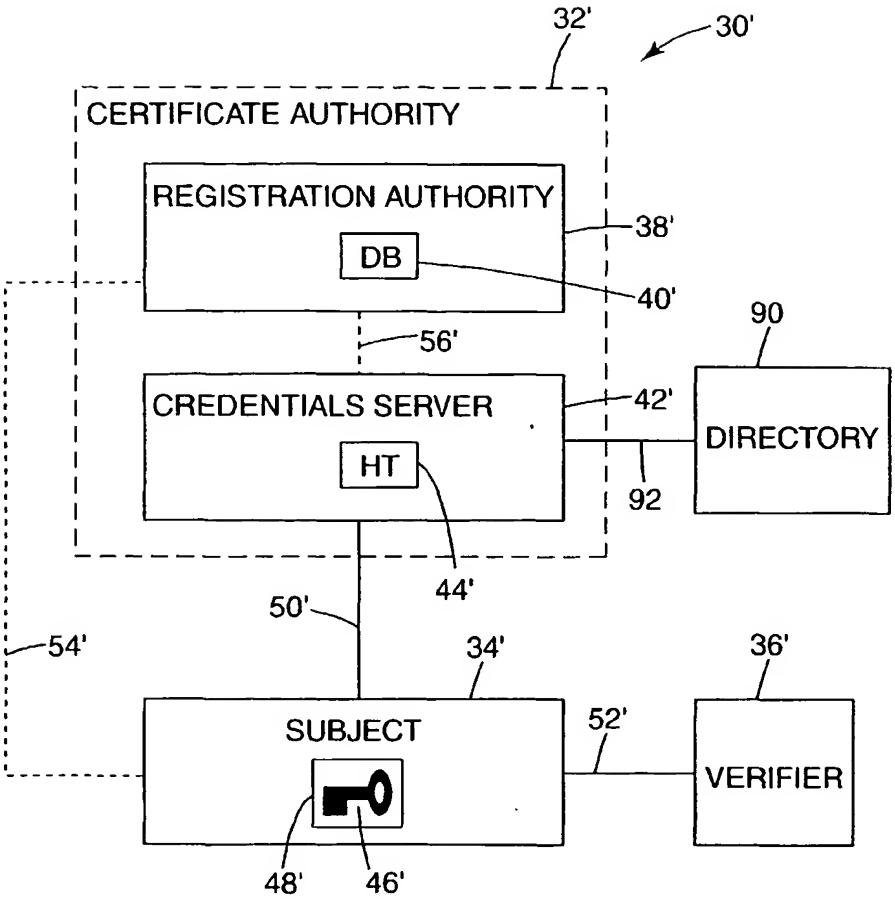
32'
30'
CERTIFICATE AUTHORITY
REGISTRATION AUTHORITY
38'
DB
40'
56'
90
CREDENTIALS SERVER
42'
DIRECTORY
HT
92
44'
50'
54'
34'
36'
SUBJECT
52'
VERIFIER
48'
46'

Fig. 8

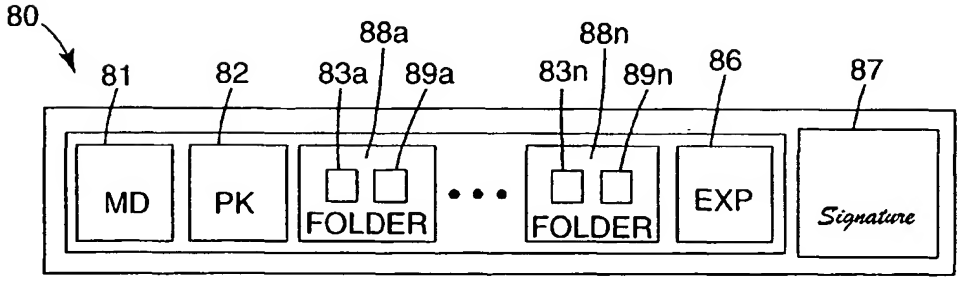
80
81
82
88a
83a
89a
88n
83n
89n
86
87
MD
PK
FOLDER
FOLDER
EXP
Signature

Fig. 9

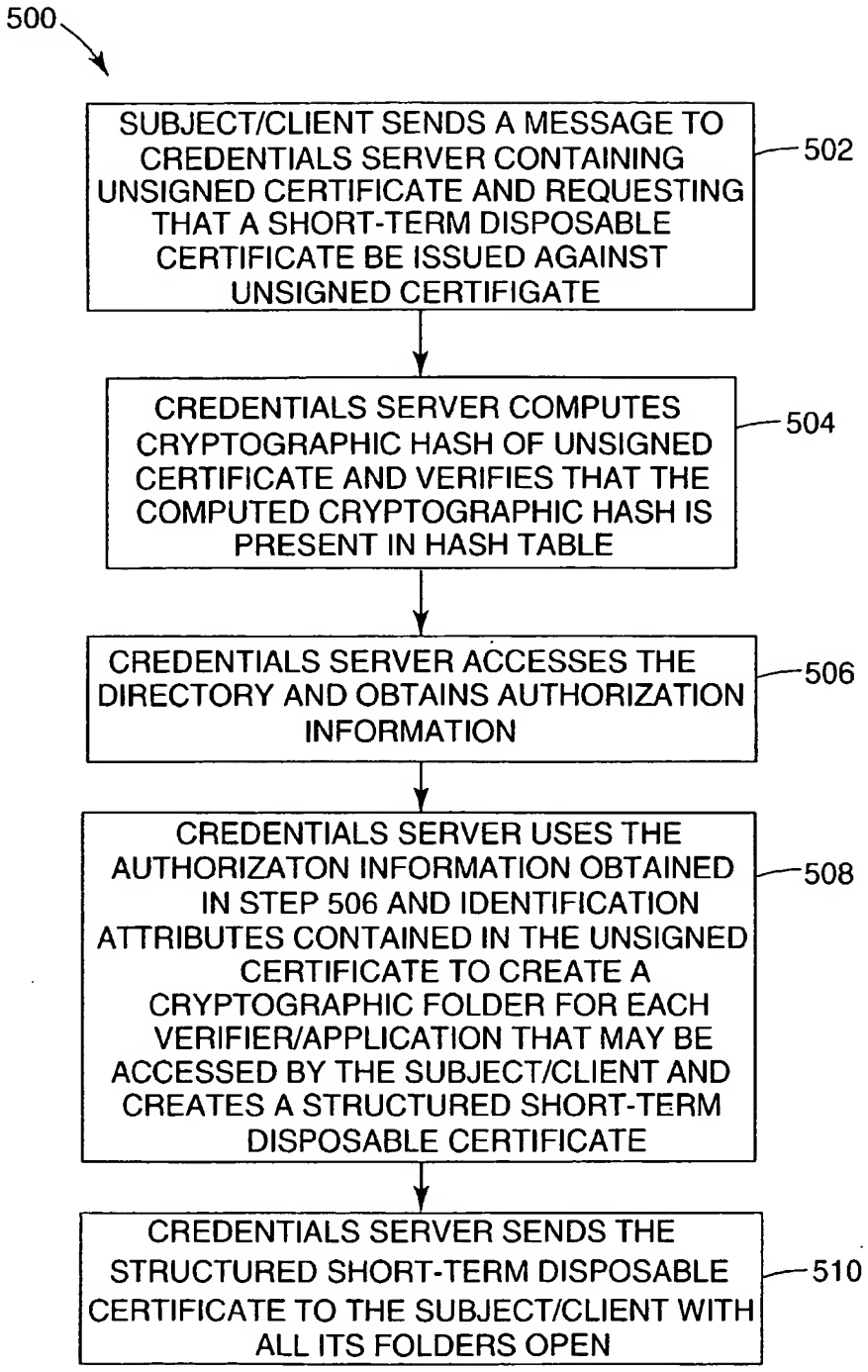
500
SUBJECT/CLIENT SENDS A MESSAGE TO CREDENTIALS SERVER CONTAINING UNSIGNED CERTIFICATE AND REQUESTING THAT A SHORT-TERM DISPOSABLE CERTIFICATE BE ISSUED AGAINST UNSIGNED CERTIFIGATE
502
CREDENTIALS SERVER COMPUTES CRYPTOGRAPHIC HASH OF UNSIGNED CERTIFICATE AND VERIFIES THAT THE COMPUTED CRYPTOGRAPHIC HASH IS PRESENT IN HASH TABLE
504
CREDENTIALS SERVER ACCESSES THE DIRECTORY AND OBTAINS AUTHORIZATION INFORMATION
506
CREDENTIALS SERVER USES THE AUTHORIZATON INFORMATION OBTAINED IN STEP 506 AND IDENTIFICATION ATTRIBUTES CONTAINED IN THE UNSIGNED CERTIFICATE TO CREATE A CRYPTOGRAPHIC FOLDER FOR EACH VERIFIER/APPLICATION THAT MAY BE ACCESSED BY THE SUBJECT/CLIENT AND CREATES A STRUCTURED SHORT-TERM DISPOSABLE CERTIFICATE
508
CREDENTIALS SERVER SENDS THE STRUCTURED SHORT-TERM DISPOSABLE CERTIFICATE TO THE SUBJECT/CLIENT WITH ALL ITS FOLDERS OPEN
510

Fig. 10

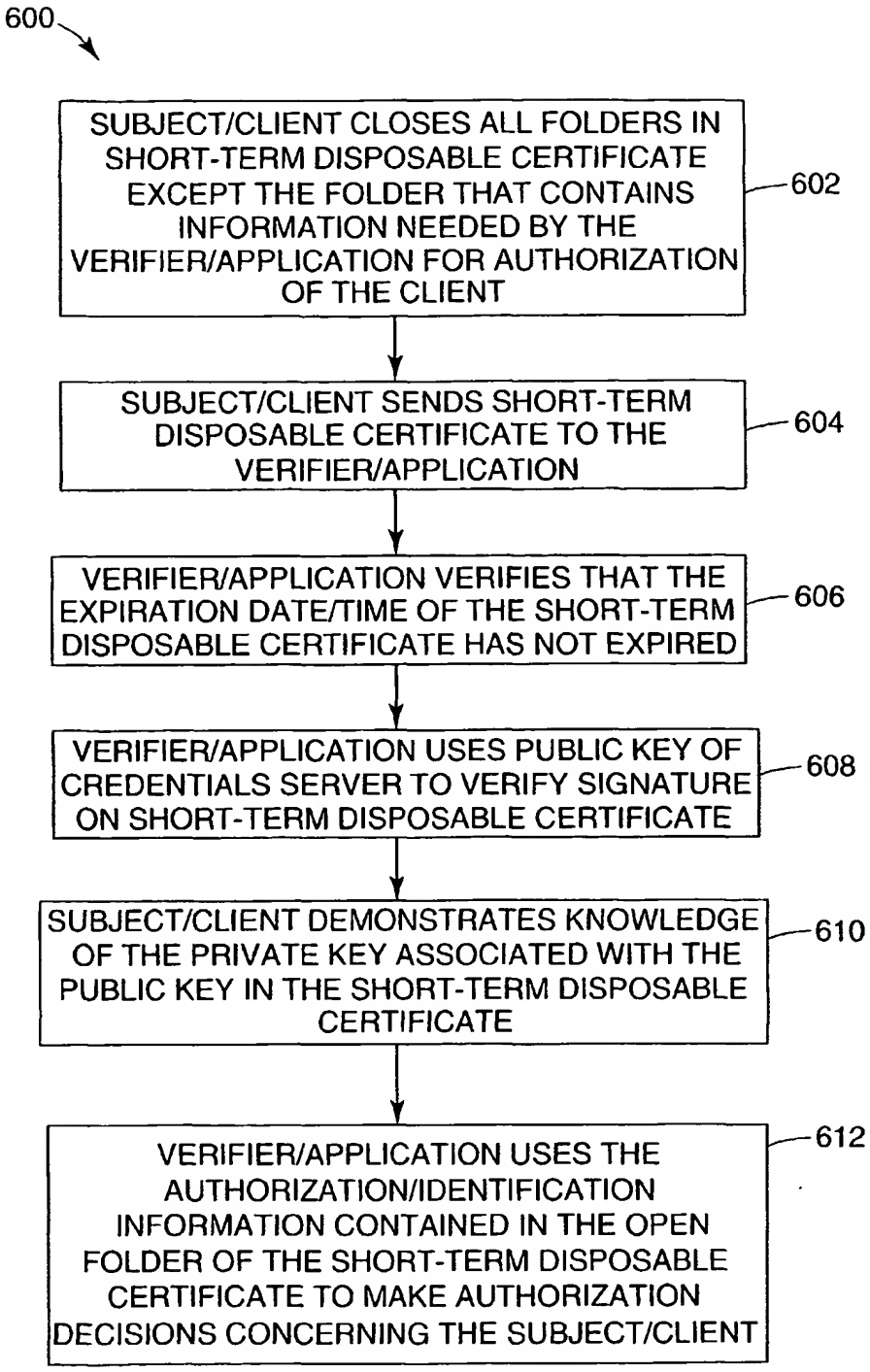
600
SUBJECT/CLIENT CLOSES ALL FOLDERS IN SHORT-TERM DISPOSABLE CERTIFICATE EXCEPT THE FOLDER THAT CONTAINS INFORMATION NEEDED BY THE VERIFIER/APPLICATION FOR AUTHORIZATION OF THE CLIENT
602
SUBJECT/CLIENT SENDS SHORT-TERM DISPOSABLE CERTIFICATE TO THE VERIFIER/APPLICATION
604
VERIFIER/APPLICATION VERIFIES THAT THE EXPIRATION DATE/TIME OF THE SHORT-TERM DISPOSABLE CERTIFICATE HAS NOT EXPIRED
606
VERIFIER/APPLICATION USES PUBLIC KEY OF CREDENTIALS SERVER TO VERIFY SIGNATURE ON SHORT-TERM DISPOSABLE CERTIFICATE
608
SUBJECT/CLIENT DEMONSTRATES KNOWLEDGE OF THE PRIVATE KEY ASSOCIATED WITH THE PUBLIC KEY IN THE SHORT-TERM DISPOSABLE CERTIFICATE
610
VERIFIER/APPLICATION USES THE AUTHORIZATION/IDENTIFICATION INFORMATION CONTAINED IN THE OPEN FOLDER OF THE SHORT-TERM DISPOSABLE CERTIFICATE TO MAKE AUTHORIZATION DECISIONS CONCERNING THE SUBJECT/CLIENT
612

Fig. 11

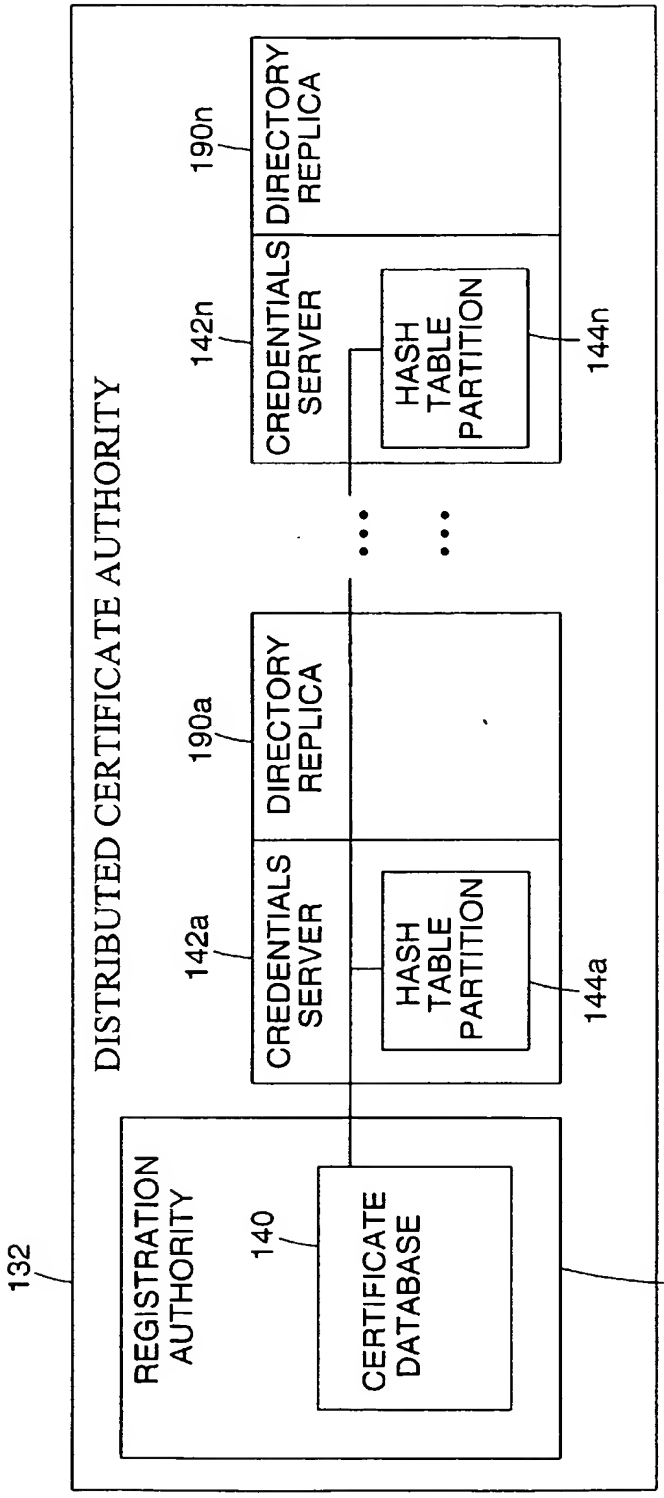
DISTRIBUTED CERTIFICATE AUTHORITY
132
REGISTRATION AUTHORITY
140
CERTIFICATE DATABASE
138
142a
CREDENTIALS SERVER
HASH TABLE PARTITION
144a
190a
DIRECTORY REPLICA
142n
CREDENTIALS SERVER
HASH TABLE PARTITION
144n
190n
DIRECTORY REPLICA

Fig. 12

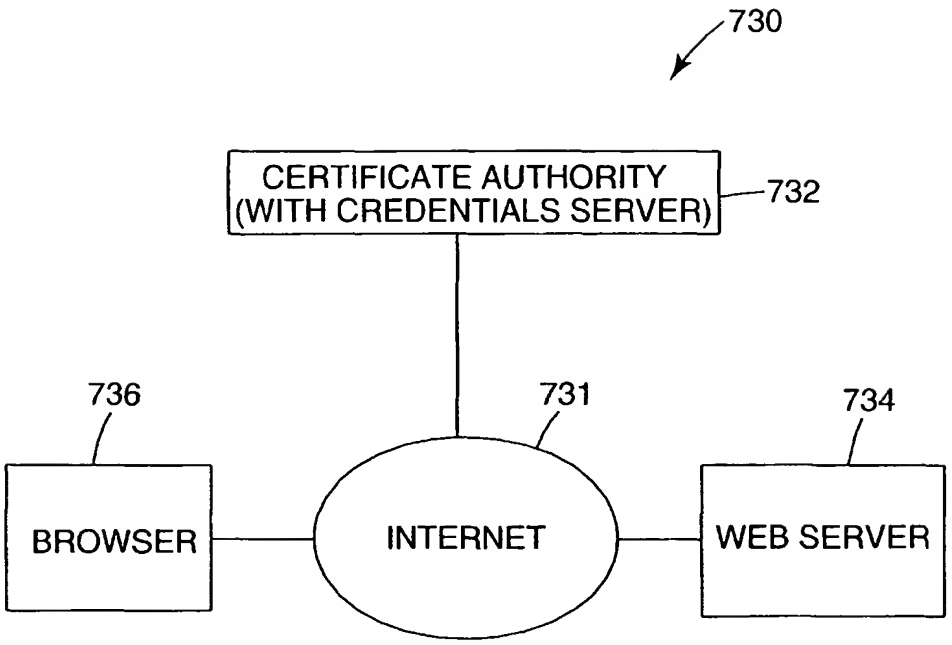
730
CERTIFICATE AUTHORITY
(WITH CREDENTIALS SERVER)
732
736
731
734
BROWSER
INTERNET
WEB SERVER

Fig. 13

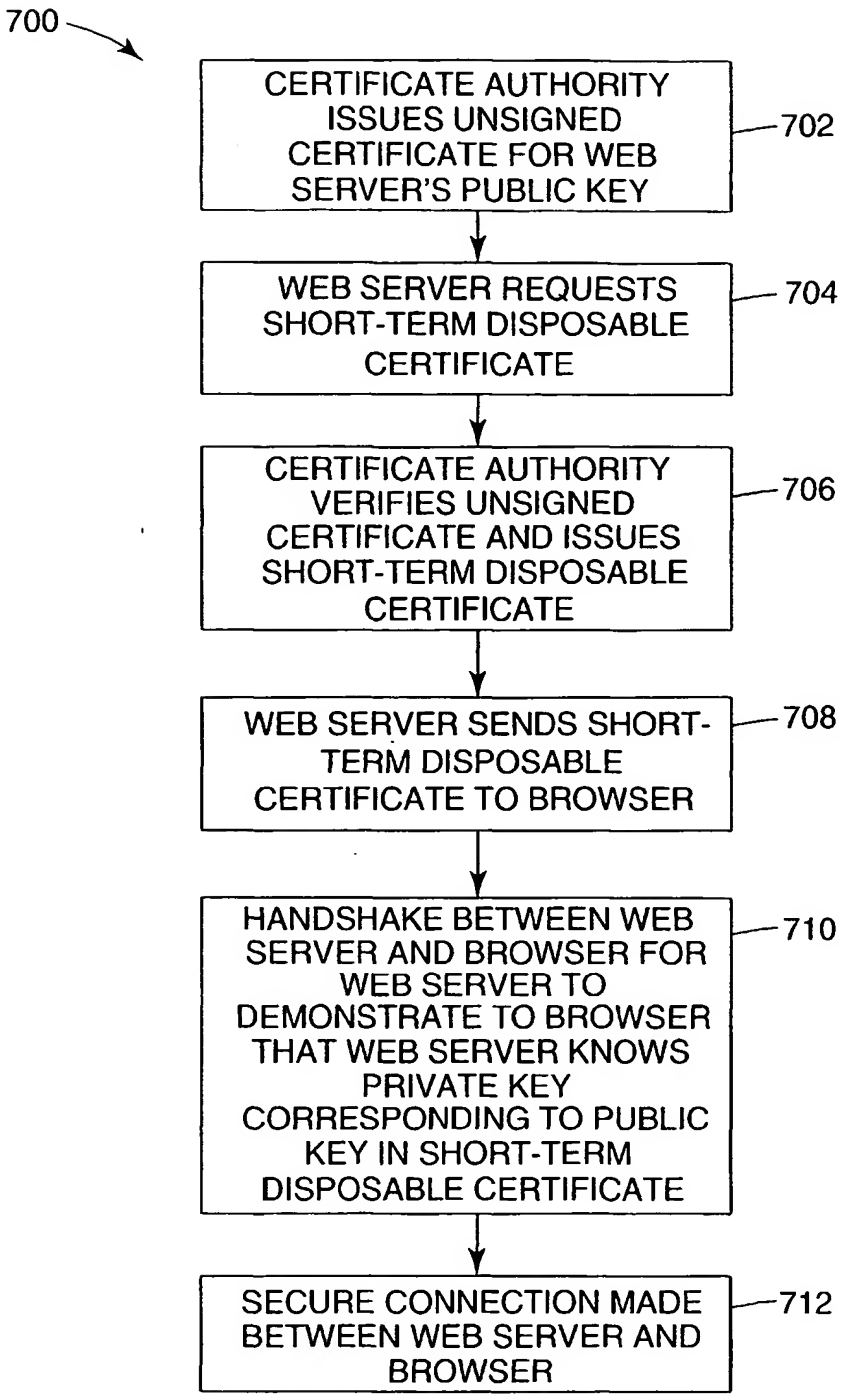
700
CERTIFICATE AUTHORITY ISSUES UNSIGNED CERTIFICATE FOR WEB SERVER'S PUBLIC KEY
702
WEB SERVER REQUESTS SHORT-TERM DISPOSABLE CERTIFICATE
704
CERTIFICATE AUTHORITY VERIFIES UNSIGNED CERTIFICATE AND ISSUES SHORT-TERM DISPOSABLE CERTIFICATE
706
WEB SERVER SENDS SHORT-TERM DISPOSABLE CERTIFICATE TO BROWSER
708
HANDSHAKE BETWEEN WEB SERVER AND BROWSER FOR WEB SERVER TO DEMONSTRATE TO BROWSER THAT WEB SERVER KNOWS PRIVATE KEY CORRESPONDING TO PUBLIC KEY IN SHORT-TERM DISPOSABLE CERTIFICATE
710
SECURE CONNECTION MADE BETWEEN WEB SERVER AND BROWSER
712

Fig. 14

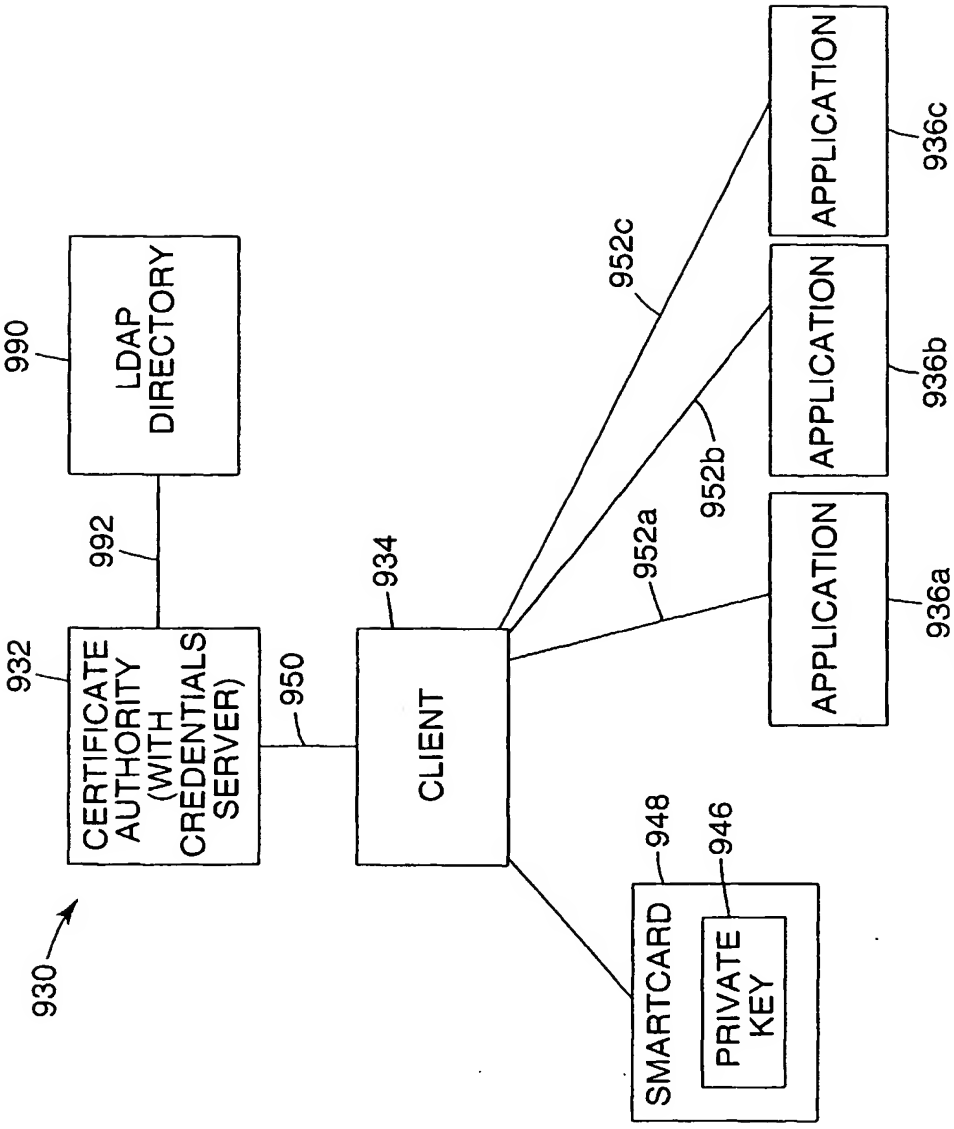
930
CERTIFICATE AUTHORITY (WITH CREDENTIALS SERVER)
932
992
LDAP DIRECTORY
990
950
CLIENT
934
952a
952b
952c
APPLICATION
936a
APPLICATION
936b
APPLICATION
936c
SMARTCARD
948
PRIVATE KEY
946

Fig. 15

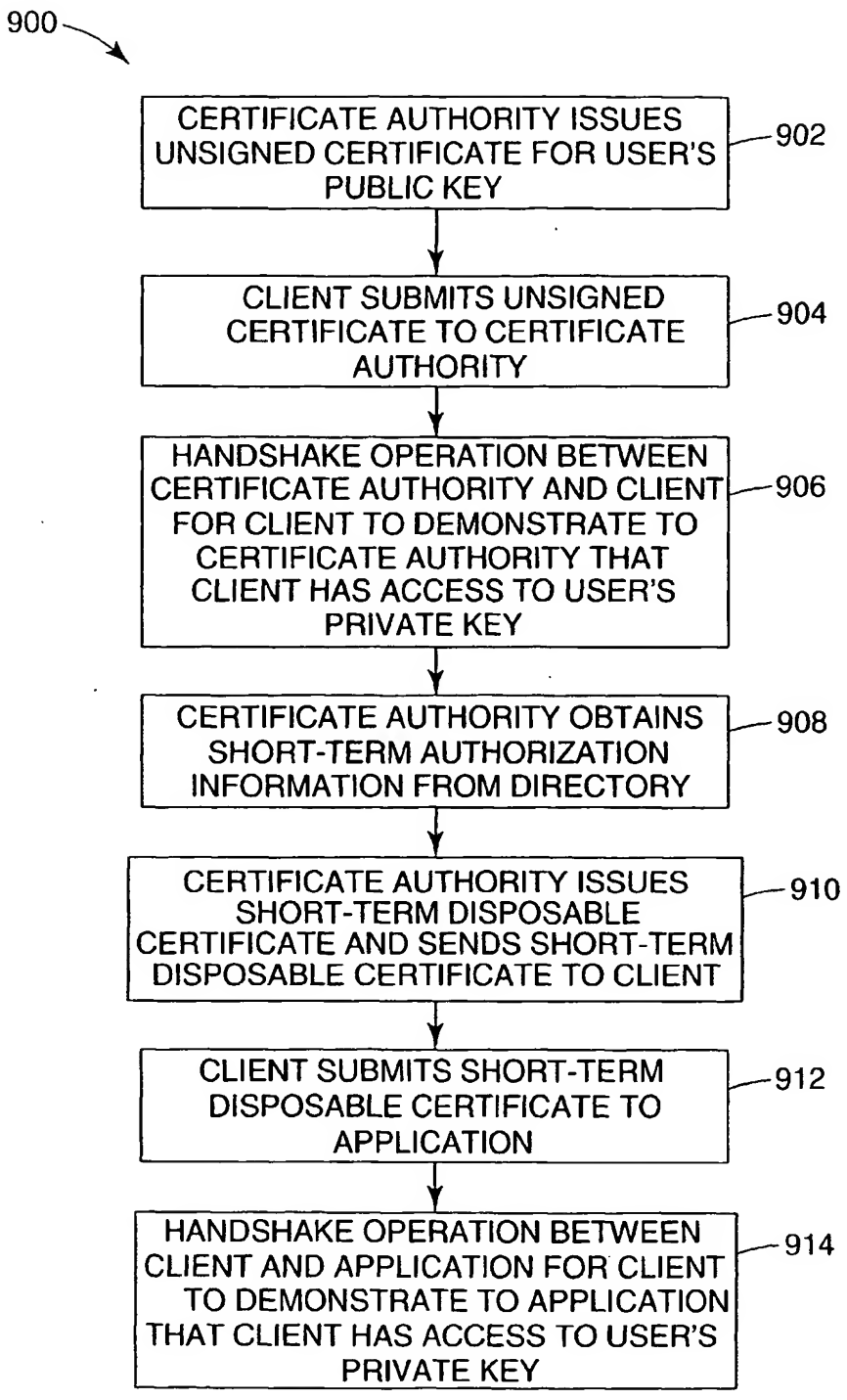
900
CERTIFICATE AUTHORITY ISSUES UNSIGNED CERTIFICATE FOR USER'S PUBLIC KEY
902
CLIENT SUBMITS UNSIGNED CERTIFICATE TO CERTIFICATE AUTHORITY
904
HANDSHAKE OPERATION BETWEEN CERTIFICATE AUTHORITY AND CLIENT FOR CLIENT TO DEMONSTRATE TO CERTIFICATE AUTHORITY THAT CLIENT HAS ACCESS TO USER'S PRIVATE KEY
906
CERTIFICATE AUTHORITY OBTAINS SHORT-TERM AUTHORIZATION INFORMATION FROM DIRECTORY
908
CERTIFICATE AUTHORITY ISSUES SHORT-TERM DISPOSABLE CERTIFICATE AND SENDS SHORT-TERM DISPOSABLE CERTIFICATE TO CLIENT
910
CLIENT SUBMITS SHORT-TERM DISPOSABLE CERTIFICATE TO APPLICATION
912
HANDSHAKE OPERATION BETWEEN CLIENT AND APPLICATION FOR CLIENT TO DEMONSTRATE TO APPLICATION THAT CLIENT HAS ACCESS TO USER'S PRIVATE KEY
914

Fig. 16

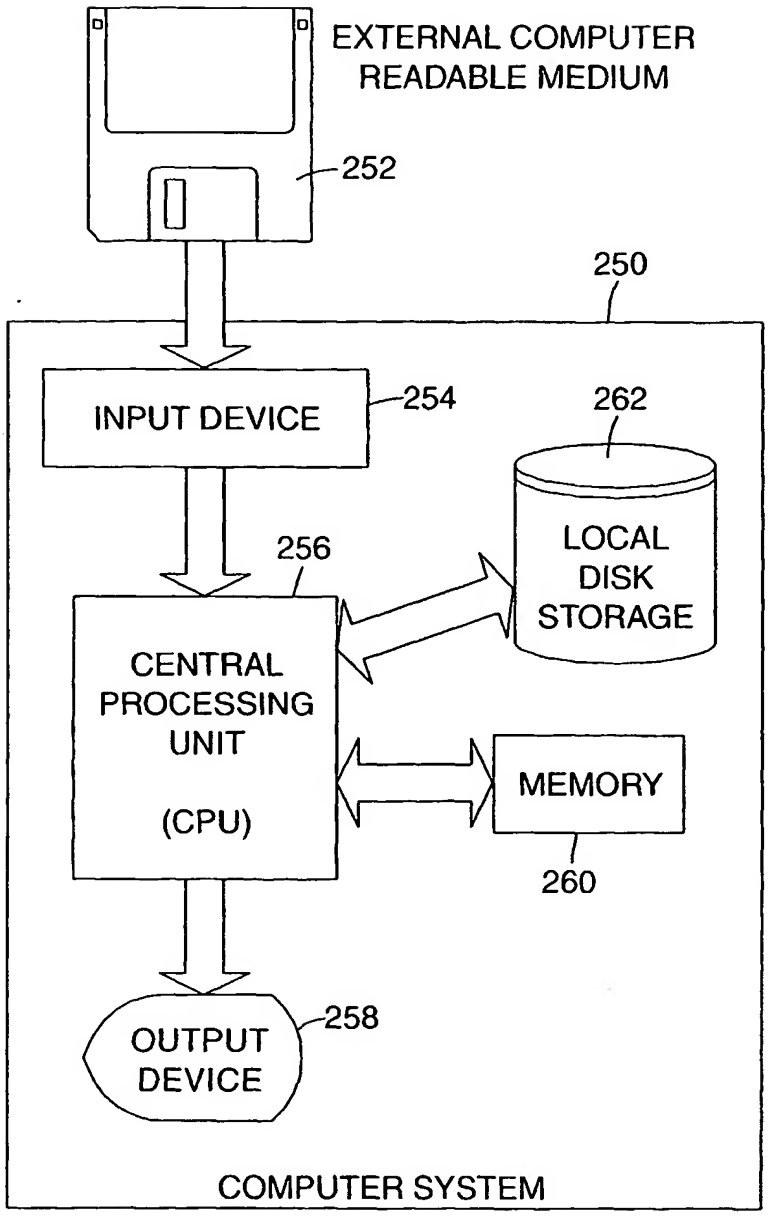
EXTERNAL COMPUTER READABLE MEDIUM
252
250
INPUT DEVICE
254
262
LOCAL DISK STORAGE
256
CENTRAL PROCESSING UNIT
(CPU)
MEMORY
260
OUTPUT DEVICE
258
COMPUTER SYSTEM

Fig. 17

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 117 207 A3

(12) EUROPEAN PATENT APPLICATION

(88) Date of publication A3:
09.10.2002 Bulletin 2002/41

(43) Date of publication A2:
18.07.2001 Bulletin 2001/29

(21) Application number: **01300224.1**

(22) Date of filing: **11.01.2001**

(51) Int Cl.[7]: **H04L 9/32**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE TR
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **14.01.2000 US 483356**

(71) Applicant: **Hewlett-Packard Company**
Palo Alto, California 94304-1112 (US)

(72) Inventor: **Corella, Francisco**
Hayward, California 94541 (US)

(74) Representative: **Powell, Stephen David et al**
WILLIAMS, POWELL & ASSOCIATES,
4 St Paul's Churchyard
London EC4M 8AY (GB)

(54) Public key infrastructure

(57) A PKI (30) includes an off-line registration authority (38) that issues a first unsigned certificate (60) to a subject (34) that binds a public key (62) of the subject to long-term identification information (63) related to the subject and maintains a certificate database (40) of unsigned certificates in which it stores the first unsigned certificate. An on-line credentials server (42) issues a short-term disposable certificate (70) to the subject that binds the public key of the subject from the first unsigned certificate to the long-term identification information related to the subject from the first unsigned certificate. The credentials server maintains a table (44) that contains entries corresponding to valid unsigned certificates stored in the certificate database. The subject presents the short-term disposable certificate to a verifier (36) for authentication and demonstrates that the subject has knowledge of a private key corresponding to the public key (46) in the short-term disposable certificate.

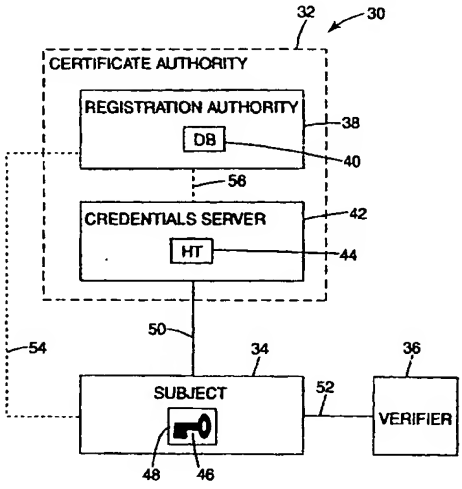

Fig. 1

EP 1 117 207 A3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 01 30 0224

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
A	WO 99 35783 A (CYBERSAFE CORP) 15 July 1999 (1999-07-15) * abstract * * page 5, line 27 - page 6, line 8 * * page 10, line 12 - page 12, line 21 * * page 16, line 6 - page 17, line 2 *	1-4,8,9, 12, 14-17, 19,20,22	H04L9/32
A	US 5 982 898 A (HSU YUNG-KAO ET AL) 9 November 1999 (1999-11-09) * abstract * * column 4, line 3 - column 8, line 21 * * figures 2,3 *	1-4,8,9, 12, 14-17, 19,20,22 6	
A	WO 99 19845 A (AT & T CORP) 22 April 1999 (1999-04-22) * abstract * * page 8, line 18 - page 14, line 23 *	1,4,8,9, 14,17, 19,20	TECHNICAL FIELDS SEARCHED (Int.Cl.7) H04L G06F
A	MENEZES ET AL: "HANDBOOK OF APPLIED CRYPTOGRAPHY" HANDBOOK OF APPLIED CRYPTOGRAPHY, CRC PRESS SERIES ON DISCRETE MATHEMATICES AND ITS APPLICATIONS, BOCA RATON, FL, CRC PRESS, US, 1997, XP002173298 ISBN: 0-8493-8523-7 * page 39, line 1 - line 38 * * page 559, line 7 - page 560, line 38 * * page 576, line 18 - page 577, line 37 *	1-4,8	

-/--

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	16 August 2002	Dujardin, C

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 01 30 0224

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
A	RIVEST R L: "CAN WE ELIMINATE CERTIFICATE REVOCATION LISTS?" FINANCIAL CRYPTOGRAPHY. INTERNATIONAL CONFERENCE, XX, XX, February 1998 (1998-02), pages 178-183, XP000997964 * the whole document *	1-4,8	
			TECHNICAL FIELDS SEARCHED (Int.Cl.7)

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	16 August 2002	Dujardin, C

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 01 30 0224

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

16-08-2002

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9935783 A	15-07-1999	AU 2452699 A	26-07-1999
		CA 2313328 A1	15-07-1999
		EP 1042885 A1	11-10-2000
		JP 2002501218 T	15-01-2002
		WO 9935783 A1	15-07-1999
US 5982898 A	09-11-1999	WO 9839878 A1	11-09-1998
WO 9919845 A	22-04-1999	BR 9806293 A	18-09-2001
		EP 0941526 A1	15-09-1999
		WO 9919845 A1	22-04-1999
		US 6125349 A	26-09-2000

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82